FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Uma abordagem ontológica para modelação de informação espaciotemporal com aplicações em transportes

Alexey Seliverstov

DISSERTAÇÃO

Mestrado Integrado em Engenharia Informática e Computação

Orientador: Rosaldo Rossetti

Junho 2015

Uma abordagem ontológica para modelação de informação espaciotemporal com aplicações em transportes

Alexey Seliverstov

Mestrado Integrado em Engenharia Informática e Computação

Junho 2015

Resumo

O trabalho realizado na dissertação é desenvolvido no contexto da análise de dados espacio-temporais, mais propriamente dados de redes rodoviárias.

Nos dias de hoje existe uma grande quantidade de dados. Estes estão constantemente a ser gerados por todos ou quase todos os dispositivos eletrónicos que nos rodeiam. O aproveitamento destes dados tem cada vez mais atraído a atenção de investigadores e de empresas que investigam e investem nesse sentido.

Alguns desses dados são gerados, por exemplo, por sensores instalados nas estradas. Estes tanto podem ser fixos, como é o caso das espiras magnéticas ou câmaras de videovigilância, como também podem ser móveis, o que acontece no caso dos *floating cars*. Todos estes sensores geram dados e a sua leitura, compreensão e uso são muito importantes para a realização de uma boa análise acerca de tráfego rodoviário.

Para que seja possível analisar dados provenientes de variadas fontes, ou neste caso oriundos de variados sensores, é necessário efetuar junção de dados ou até das próprias fontes de dados. Isso é uma tarefa complexa e tediosa se for feita com base em técnicas clássicas como agregação de esquemas relacionais ou até junção manual dos dados. É necessário arranjar técnicas que permitam efetuar essa tarefa de uma maneira fácil e de preferência automática.

Felizmente métodos com base em ontologias surgiram para resolver este tipo de problemas. Com estes métodos é possível juntar várias fontes de dados com a manutenção do significado dos dados e sem a necessidade de efetuar tarefas manualmente.

Com base nesta metodologia, pretende-se neste trabalho abordar ontologicamente tanto os dados como os sensores rodoviários, integrando-os numa única representação. Desta forma, vários clientes podem ter fácil acesso a estes dados, sendo o resultado desta integração disponibilizado através de uma arquitetura orientada a serviços.

A povoação do modelo ontológico é feita inicialmente com registos (ou *logs* em inglês) GPS, dados de espiras magnéticas (dados da VCI cedidos pela Armis) e dados vetoriais de *OpenStreetMaps*. Como trabalho futuro poderão ser utilizados dados provenientes de simuladores como o SUMO para o preenchimento de eventuais lacunas que possam existir nos dados reais. Também é pretendido alargar o suporte a outros simuladores.

Como resultado final elaborou-se um repositório geográfico científico sob o nome Trontegra, proveniente das palavras **Tr**ansportes, **On**tologia e In**tegra**ção. Este sistema serve para efetuar junção de fontes de dados por meio de ontologia e permite a serviços executar as suas *queries* através de um ponto de acesso que este providencia. Trontegra pode ser usado por variados utilizadores, com diferentes interesses ou objetivos relativamente aos dados disponibilizados. Depois de ter realizado testes à prova de conceito criada, concluiu-se que o seu uso é mais focado em serviços que não necessitam de dados em tempo real devido ao *overhead* adicionado por Trontegra na execução das *queries*.

Abstract

This dissertation work is developed having as the context the spatiotemporal data analysis, specifically transportation data.

Nowadays there is too much data available. This is because almost every electronic device that is around us generate data. Usage of this resource attracts researchers and companies to research and invest on ways to do so. Even roads are generating data through its sensors. These sensors can be fixed, which is the case of inductive loops and video surveillance cameras, and mobile, as it is the case of floating cars. Its data readability, comprehension and usage is of vital importance for a good traffic analysis.

To be possible to analyse data from different data sources, or sensors in this specific case, it is necessary to integrate those into a single point of access. Having this in mind, it is also known that this task is difficult if done manually. So it is necessary to research new techniques which would allow us to simplify this task, and preferentially automate it.

Fortunately there are methods based on ontologies which help solving those problems. They allow us to integrate many data sources and maintain its original meaning and form. Based on this methodology it is intended to approach ontologically these sensors and their data, integrating them into a single representation. After that, it is needed to share this integrated system through a service oriented architecture. This will allow clients to easily access all data present in our system.

GPS logs, vector information from OpenStreetMaps and inductive-loop data (the last provided by the Armis Company) is used to populate this ontological model. Afterwards, as future work, data from simulators such as SUMO will be integrated so as to fill in possible gaps that might not be covered by real data.

As a result of this dissertation, a scientific geographical repository has been specified and implemented, called Trontegra (formed from 3 words: **Tr**ansportation, **On**tology and In**tegra**tion). This system integrates many datasources into one ontological model through which one service can query data present in the system. This data can be used for transportation analysis by many clients with different needs and interests. After testing the created proof of concept, it was concluded that it will mostly be used by services that do not need data in real time because of the overhead that Trontegra adds to the queries.

Agradecimentos

Desde o início agradeço a Deus por me ter feito tal como eu sou, me ter dado vida com tudo aquilo que eu necessito e não o que eu quero, e me ter salvo através da sua morte e ressurreição. Agradeço também à minha família que sempre me apoiou e me guiou, quaisquer que fossem as circunstâncias.

Igualmente estou grato à minha namorada Sofia que esteve sempre comigo, nos momentos altos e baixos da minha vida e me incentivou a continuar, a lutar e a avançar.

Agradeço também ao professor Rosaldo Rossetti que me introduziu a este tema tratado na dissertação, me ajudou com boa orientação e me guiou na construção desta. Agradeço também pela sua amizade demonstrada para com os seus alunos, sempre com boa disposição e pronto a ajudar.

Fico também eternamente grato aos meus colegas, mais especificamente o Joaquim Barros, Tiago Azevedo, Filipe Oliveira e Tiago Costa, sem os quais o meu percurso académico não seria tão risonho e gratificante.

Alexey Seliverstov

"Haja luz", e houve luz. ***Génesis 1:3***
"Eu sou a luz do mundo; quem me segue
de modo algum andará em trevas, mas terá a luz da vida." ***João 8:12***
"Em verdade, em verdade vos digo que
quem ouve a minha palavra, e crê naquele que
me enviou, tem a vida eterna e não entra em juízo,
mas já passou da morte para a vida." ***João 5:24***

Jesus Cristo

Conteúdo

Lista de Figuras

Lista de Tabelas

Abreviaturas e Símbolos

API	Application Programming Interface
WWW	*World Wide Web*
GIS	Geographic Information System
GIS-T	Geographic Information System for Transportation
XML	eXtensible Markup Language
OWL	Ontology Web Language
RDF	Resource Description Framework
W3C	World Wide Web Consortium
URI	Uniform resource identifier
GSM	Global System for Mobile Communications
GPRS	General packet radio service
QoS	Quality of Service
SPARQL	SPARQL Protocol and RDF Query Language
SGBD	Sistema Gestor de Base de Dados
RAM	Random Access Memory
MB	Megabytes
GB	Gigabytes

Capítulo 1

Introdução

Este é o capítulo introdutório deste trabalho. Aqui vai ser explicado qual é o problema que se pretende resolver, incluindo também o seu contexto e a motivação para tal. Serão igualmente expostos os objetivos que se pretendem alcançar na resolução do problema e posteriormente será apresentada a estrutura do relatório.

1.1 Contexto

Hoje em dia, a realização da análise de dados gerados por tráfego rodoviário é essencial para o bom funcionamento e desenvolvimento das estradas portuguesas. Devido à atual possibilidade de existirem sensores instalados nas estradas e não só, existem grandes quantidades de dados a serem gerados. Isto permite aos analistas de tráfego saber em tempo real o estado de trânsito rodoviário numa determinada área geográfica sem a necessidade de lá estar ou para lá se deslocar.

O processamento deste conhecimento somente é possível se for feita uma aglomeração de dados provenientes dos mais variados sensores. Estes poderão ser fixos num determinado ponto da faixa de rodagem mas também têm a capacidade de serem móveis. No caso de sensores fixos estes poderão ser câmaras de vídeo vigilância, espiras magnéticas, tubos pneumáticos, entre outros [KMG06]. No caso de sensores móveis, o funcionamento destes consiste geralmente no fornecimento de coordenadas GPS de um veículo equipado com este sensor, como é o caso de *floating cars* (carros flutuantes) que têm exatamente isso como sua base de funcionamento [STBW02].

Para efetuar a operação de junção de dados de variadas fontes atualmente são necessárias ferramentas informáticas complexas. Estas terão de ser capazes de receber os dados dos sensores, limpar de possíveis erros que possam ter surgido no processo de recolha, e posteriormente armazená-los para mais tarde serem apresentados aos analistas. Estes com base nisso poderão tirar conclusões e resultados acerca do estado do trânsito numa determinada zona geográfica.

1.2 Motivação e Objetivos

Devido à utilidade dos dados gerados pelos sensores rodoviários e das conclusões possivelmente tiradas a partir da análise destes, a junção de dados gerados por vários sensores num único ambiente seria bastante favorável para uma análise rodoviária mais completa [RB99]. Para que isso seja possível é necessário aplicar métodos de junção de informação aos dados provenientes dos mais variados sensores. Existem diversos métodos para efetuar esta junção de dados com base na área da inteligência artificial, tais como reconhecimento de padrões, estimativas estatísticas [EFLK11] e o método ontológico. É no último sobre o qual este trabalho se debruçará.

Na atualidade, a integração de dados ou até de fontes de dados é possível de ser feita aplicando táticas tradicionais de junção de dados através de mapeamentos e de correspondência de esquemas, o que no entanto deixa bastantes conflitos heterogéneos na informação devido à falta de descrição semântica desta. Isto dificulta uma análise multidisciplinar precisa e limita a possibilidade de serem tiradas conclusões mais complexas e completas por parte de vários especialistas de várias áreas pois os dados estão dispersos e não tem conexões com outros dados de outras fontes de informação. Daí surge a necessidade de arranjar um processo que consiga centralizar a junção de dados de uma maneira clara e concisa para uma posterior utilização destes por um serviço que o queira fazer, facilitando também a adição de novas fontes de dados (p.ex. um sensor novo).

Tendo isto em conta, como objetivo deste trabalho pretende-se contribuir para a solução do problema da dificuldade de junção de dados provenientes de variadas fontes, através da aplicação de um modelo ontológico sobre os dados e suas fontes. Como resultado final tem-se um sistema capaz de integrar variadas fontes de dados para um modelo ontológico genérico. Este servirá como base do modelo de dados gozando das vantagens que as ontologias trazem na parte da semântica e explicação de conceitos através de relações, diminuindo assim a possibilidade de existirem conflitos nos dados que forem juntos. Para facilitar o acesso aos dados é pretendido também aplicar à solução uma arquitetura orientada a serviços. Desta forma, outras entidades que pretendam utilizar os dados poderão aceder facilmente efetuar as consultas que se pretendem realizar conforme a existente descrição das fontes de dados na ontologia geral.

De modo a generalizar para um objetivo mais abstrato, é de indicar que pretende-se desenvolver um Repositório Geográfico Científico baseado em noções *Linked Data*, explicados na Secção 2.4. Esta ferramenta possibilitará a consulta de dados de variadas fontes espaciotemporais através de um só ponto de acesso global.

1.3 Questões de Investigação

Com base no método descrito em [AO05] e aplicado no contexto deste trabalho em A.1.1, foram elaboradas as perguntas a seguir apresentadas. Para se poder responder às questões foi feito um levantamento do estado de arte com base na pesquisa em várias bases de dados eletrónicas

relacionadas com a área da engenharia. É de notar que nem todas as perguntas poderão ser diretamente respondidas a partir da literatura que se encontrou, sendo que essa parte será a contribuição deste trabalho à comunidade científica.

1. Como interoperabilizar múltiplas perspetivas de análise de transporte/tráfego a partir de *Open Data*?
2. Qual o método de junção de fontes de dados a utilizar para uma eficiente obtenção e processamento de dados rodoviários (*data integration*; *sensor integration*)?
3. Quais os métodos de junção de fontes de dados, para a questão acima, tendo por base um modelo ontológico?
4. Como automatizar o processo de adição de uma nova fonte de informação ao já existente modelo ontológico?
5. De que forma será possível aplicar uma arquitetura orientada a serviços ao modelo ontológico que se pretende aplicar, de modo a permitir outras entidades acederem aos dados?
6. Quão eficientes são as interrogações num sistema baseado em integração por ontologia?

1.4 Estrutura do Relatório

As partes seguintes do artigo estão organizadas da seguinte forma: no Capítulo 2 está presente uma revisão de conceitos usados no âmbito deste trabalho. No Capítulo 3 estão explicitados trabalhos e ferramentas relevantes que servirão de base para a elaboração deste trabalho. No Capítulo 4 é descrito mais a fundo o problema e a solução proposta para o resolver. É também incluído um subcapítulo que demonstra alguns casos de utilização deste sistema. No Capítulo 5 são discutidos 3 tópicos importantes: a implementação de Trontegra, que inclui a sua arquitetura, as tecnologias e as ferramentas utilizadas; são também feitos testes e analisados os seus resultados efetuados à eficiência da ferramenta desenvolvida; e por último é feita uma discussão acerca da aplicabilidade da ferramenta no âmbito dos dados espaciotemporais. Finalmente no Capítulo 6 é feito o resumo de toda a dissertação, é analisada a satisfação dos objetivos desta e é feita uma análise aos possíveis trabalhos futuros ou melhorias deste projeto.

Capítulo 2

Revisão de Conceitos

Neste capítulo irão ser explicados os conceitos e o vocabulário que irão ser posteriormente utilizados nesta dissertação. Para isso, será analisada literatura relacionada com cada um dos tópicos e serão citadas as fontes a partir das quais cada conceito foi retirado. Será também indicado onde cada um dos conceitos poderá ser aplicado de maneira a esclarecer melhor a sua relação com este trabalho.

2.1 *GIS*

Geographic Information Systems (**GIS**) ou Sistemas de Informação Geográfica (em Português) são sistemas de informação com capacidade de analisar, armazenar e representar de uma maneira complexa dados com atributos espaciais [NB01] e por causa disso mesmo estes são diferentes dos sistemas tradicionais [Due79, SMSE87, MGR91]. Por causa dessas capacidades estes sistemas são agora usados numa variedade de áreas que envolvem resolução de problemas relacionados com dados espaciais [SE90].

Sendo transportes a área de aplicação deste trabalho, a adaptação de *GIS* a estes dados é **GIS-T**. GIS-T ou Sistemas de Informação Geográfica em Transportes herda todas as propriedades de *GIS* e usa uma combinação da informação relativa ao tráfego com dados presentes no sistema geográfico. É utilizado para resolver problemas relacionados com planeamento e cálculo de rotas, gestão de veículos e auxiliar de navegação [ZCZ05], entre outros.

Em termos de aplicação prática no trabalho irá ser usada uma adaptação simplificada de GIS-T. Esta providencia ao sistema toda a informação relativa a estradas, e que é usada no contexto dos dados rodoviários disponíveis no sistema. Ou seja, por outras palavras, o sistema irá ter um mapa, que é construído pelo sistema com base nos seus dados, e povoado com informação espaciotemporal igualmente deste.

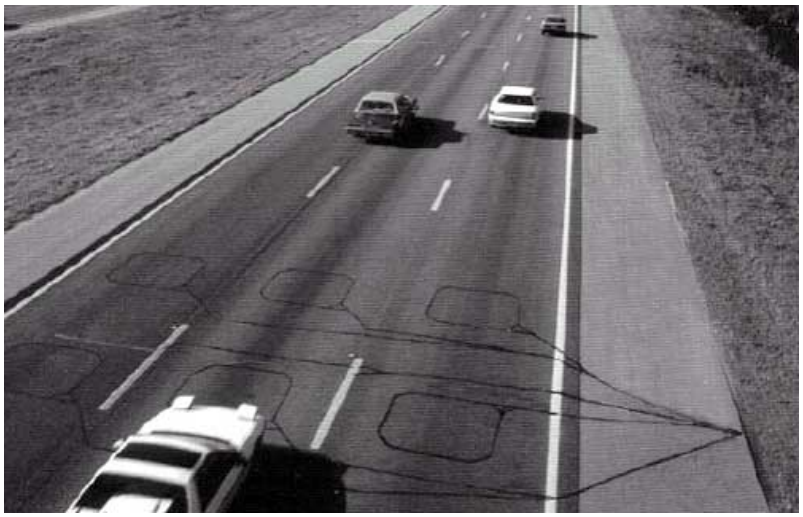

Figura 2.1: Exemplo de uma instalação de espiras magnéticas numa estrada (URL da imagem: http://ops.fhwa.dot.gov/freewaymgmt/publications/frwy_mgmt_handbook/images/fig15-1.jpg)

2.2 Sensores Rodoviários

No contexto do trabalho um sensor rodoviário é um aparelho que de alguma maneira tem a capacidade de gerar dados acerca do estado de trânsito numa determinada área em que este se encontre, instalado ou móvel. Existe uma grande variedade de sensores rodoviários dos quais somente alguns vão ser apresentados a seguir. Os sensores são úteis para este trabalho porque é através deles que são obtidos os dados.

2.2.1 Espira Magnética

Uma espira magnética, mais especificamente a que é usada nas estradas, é uma simples bobina de fio que é instalada depois de o asfalto ter sido posto na estrada, sendo que a instalação é feita perfazendo uma ranhura e pondo na mesma a bobina. Esta instalação é facilmente identificável em qualquer estrada. Geralmente funcionam por capturas durante um curto período de tempo, p.ex. 5 minutos, e os principais dados gerados são o número de carros que passaram durante o período de captura, a velocidade média dos veículos, o espaçamento entre estes, a taxa de ocupação do sensor durante aquele período (a percentagem de tempo no qual havia algum veículo por cima do sensor), a data e hora de captura. Um exemplo deste sensor é a Figura 2.1.

2.2.2 Vídeo-vigilância

Videovigilância consiste em câmaras de vídeo que se encontram instaladas nas estradas e que captam imagem. Estas capturas em tempo real permitem saber qual é o estado de trânsito na zona em que esta câmara está instalada. Um exemplo pode ser observado na Figura 2.2. Estes tipos de sensores têm sido utilizados para obtenção de dados acerca do estado de tráfego rodoviário [LRB09].

Figura 2.2: Exemplo de uma câmara de vídeo instalada para monitorizar uma auto-estrada (URL da imagem: http://www.roadtraffic-technology.com/contractor_images/pips-technology/2-MMB-Spike.jpg)

2.2.3 *Floating Cars*

Floating cars, ou *Probe Vehicles*, são veículos que usando um recetor GPS e um transmissor GSM/GPRS transmitem dados quase em tempo real do estado de trânsito a partir do local onde estão localizados. Não necessitam de recursos de instalação nas estadas pois são sensores móveis. Os principais dados gerados são a velocidade e localização do veículo e data e a hora do momento da captura. Exemplos destes sensores poderão ser autocarros, táxis ou veículos policiais [ZCWY14, ZCZ05].

Para além da caracterização do tráfego, dados de GPS podem ser utilizados para melhorar a qualidade da topologia da rede em mapas digitais [FCR09, FCR10]. Outras tecnologias também têm sido propostas para recolher dados a partir de *floating cars*, como *Bluetooth* [FRK+14].

2.3 Dados Espaciotemporais

Dados espaciotemporais na sua base permitem relacionar uma localização espacial com um instante no tempo. Em [CW11] são identificados dados como espaciotemporais aqueles aos quais foram adicionadas marcas para mostrar onde e quando estes ocorreram.

No contexto deste trabalho, estes dados são gerados pelos sensores descritos na Secção 2.2 e possuem caraterísticas rodoviárias. Outras aplicações tentam extrair dados espaciotemporais a partir de fontes não estruturadas, recorrendo a técnicas de *text mining*, por exemplo [CSR10, KFC+15].

2.4 *Open Data*

Conforme a definição em [Ope], *Open Data* (Dados Abertos em Português) são dados que podem ser livremente usados e distribuídos por qualquer pessoa, sem restrições por parte de direitos de autor ou patentes. A ideia-base de *Open Data* assenta nos mesmos conceitos dos outros projetos "Open"(Open-Source, Open Hardware, etc.). São elas:

- Disponibilidade - os dados devem estar completos e com um custo não para além do razoável, de preferência disponíveis através da Internet. Também devem estar numa forma pronta a usar e modificar.
- Uso e Distribuição - os dados devem estar disponíveis sob licenças que permitem o seu uso e distribuição.
- Participação Global - toda e qualquer pessoa tem de poder usar, reutilizar e redistribuir os dados, não podendo portanto haver qualquer tipo de discriminação tanto por parte pessoal como de grupos ou empresas.

Em termos deste trabalho, é pretendido usar Open Data como sendo também uma fonte de dados espaciotemporais com o foco para a área de transportes rodoviários.

Apesar de *Open Data* estar disponível na Web, a sua utilização e edição de maneira automática com o auxílio de máquinas é difícil caso estes não estejam explicados a nível semântico nem estejam num formato que auxilia a sua utilização automática. Para esse mesmo propósito é usado RDF para estruturar os dados de maneira a que estes possam estar disponíveis na Web para uso automático [Col14]. Depois disto, são encontradas ligações entre os dados e as várias fontes de dados. Estas conexões juntamente com os dados descritos no formado RDF formam o *Linked Open Data* - Dados Abertos e Ligados.

No caso do trabalho desta dissertação é feito exatamente isso: criação de *Linked Open Data*, onde variadas fontes de dados são ligadas umas às outras, tendo por base uma ontologia para cada uma delas e descritas no formato RDF. Isto permite com que estas possam ser interrogadas simultaneamente e estejam semanticamente descritas.

2.5 Ontologia

A definição do termo ontologia depende do contexto no qual este é aplicado, sendo que por vezes definições de várias áreas contradizem-se [NM+01]. Para o caso deste problema, irão ser consideradas várias definições que irão ser apresentadas a seguir e que juntas criam uma definição mais completa.

Uma ontologia é uma descrição formal e explícita de conceitos (também designados de classes) num domínio de interesse, das propriedades de cada classe incluindo os atributos da mesma e das restrições das propriedades [NM+01]. Representa também uma especificação explícita e

partilhável do vocabulário de conceitos e das suas relações no domínio de interesse, que por si só torna-se uma visão abstrata e simplificada do mundo [Gru93, Gag07, Wan97].

Para a construção de ontologias podem ser utilizados RDF (*Resource Description Framework*), RDFS (*Resource Description Framework Schema*), DAML (*DARPA Markup Language*) com OIL (*Ontology Interchange Layer*) ou OWL (*Ontology Web Language*). De todos OWL é o que mais se destaca pois detém uma semântica bem definida e um sistema de implementação otimizado [HC13]. OWL, atualmente na sua segunda versão é a linguagem para a construção de ontologias mais usada e à data (Fevereiro 2015) é recomendada pela W3C [W3C12]. No caso da abordagem escolhida para a elaboração da solução irá ser usado RDF pois é de mais fácil manipulação automática, apesar de OWL ser o ideal em termos semânticos. A utilização de OWL é uma questão analisada na Secção 6.4.

Conforme [W3C12] e [AJ11], a definição de OWL 2 consiste em esta ser uma linguagem ontológica para a Web Semântica com significado definido formalmente. Esta representa o domínio de interesse usando axiomas, entidades e expressões. Axiomas são suposições ou factos acerca do mundo que são tomados como verdadeiros e podem ser utilizados para deduzir outros factos através de implicações lógicas. Entidades são componentes atómicos destes axiomas que podem ser indivíduos que são representações de objetos do mundo, classes que representam conjuntos de objetos, propriedades que são relações binárias que podem relacionar dois objetos (propriedade-objeto ou *object property*) ou um objeto com um valor de dados (propriedade-dados ou *data property*), e valores dos dados que são armazenados como documentos pertencentes à Web Semântica.

As ontologias construídas com base em OWL 2 podem ser usadas juntamente com informação escrita em RDF, já que estas são maioritariamente partilhadas como documentos RDF.

Ontologias são aplicadas a variados domínios, desde as diferentes áreas de engenharia [RPKC11, HC13], de medicina [AG00, CTL12] e até da agricultura [HWSW11]. Aplicações mais recentes usam aproximações ontológicas para efetuar junção de dados provenientes de variadas fontes, tanto pertencentes a um domínio [VOL+11] como de domínios diferentes, usando para isso no último caso uma ontologia para cada domínio [JfWmWd+06].

Atualmente existem também sistemas baseados em ontologias, como é o caso de [FE99]. Neste sistema de informação geográfica utilizaram-se ontologias como sendo a base da arquitetura de dados, permitindo assim ter uma boa interoperabilidade entre as variadas fontes de dados.

No caso deste trabalho, as ontologias serão úteis para efetuar a junção de dados provenientes de variados sensores e para descrever cada um destes semanticamente.

2.5.1 Mapeamento para Ontologia

O processo de conversão de um esquema[1] para uma ontologia é designado de mapeamento [CGY07]. Este mapeamento permitirá posteriormente integrar mais facilmente as fontes de dados

[1] p.ex. esquema relacional

para um modelo geral e também executar *queries* à fonte de dados a partir da qual o mapeamento foi feito.

2.5.2 Web Semântica

A Web Semântica, conforme a definição de W3C [Wor], possibilita a existência de uma *framework* comum que permite a partilha e reutilização dos dados por comunidades, empresas e aplicações e poderão ser processados tanto automaticamente como manualmente. Esta pretende que a Web se torne num espaço composto por dados e não documentos. Estes serão acedidos usando a arquitetura geral da Web, utilizando por exemplo um URI (Identificador Uniforme de Recursos em Português) para cada objeto. As relações entre os dados deverão ser as mesmas que nos documentos atualmente.

2.5.3 *Internet of Things*

Internet of Things, ou Internet das Coisas em Português, é uma extensão da Web Semântica onde objetos e serviços interagem autonomamente entre si. Essa interação é efetuada num ambiente em que estão embutidos sistemas de informação e comunicação impercetíveis para os utilizadores. Haverá também partilha de uma semântica comum e de esquemas de endereçamento para que os agentes, possam estes ser humanos ou virtuais, interagem entre si de maneira a resolver problemas dinâmicos e complexos juntos. [GBMP13, TRC+]

2.6 SOA - Arquitetura Orientada a Serviços

SOA é uma arquitetura de construção de software que ao disponibilizar serviços através de interfaces visíveis publicamente interage com utilizadores ou com outros serviços. O modo básico de funcionamento desta arquitetura consiste na troca de mensagens feita entre os clientes, que têm o papel de requerente de um serviço, e o fornecedor do serviço. O último é responsável pela correta publicação e descrição do serviço que fornece [PvdH03].

2.7 Sumário

Neste capítulo são revistos os conceitos principais que são usados ao longo de todo este documento. A sua ordenação é propositada, partindo de conceitos mais genéricos e focando em noções mais específicas e relevantes para o trabalho em questão.

A definição deste sistema consiste em angariar variados sensores e seus dados espaciotemporais baseando-se numa ontologia. Através disto, é pretendido criar *Linked Open Data*, disponível livremente para todos os utilizarem. Como é pretendido que sejam serviços os utilizadores mais frequentes deste sistema, a distribuição dos dados é feita com base numa Arquitetura Orientada a Serviços.

Capítulo 3

Trabalhos Relacionados

Neste capítulo são analisados os trabalhos que estão relacionados com o contexto deste documento. Também pretende-se de certa forma justificar com base na pesquisa elaborada da literatura relacionada que o problema abordado nesta dissertação é realmente uma lacuna existente, em estudo pela comunidade científica atual.

3.1 Junção de Dados - Métodos Não Ontológicos

Nesta secção irá ser dada uma revisão aos métodos de junção de dados não baseados em ontologia existentes na literatura encontrada. Para isso, foi elaborada a Tabela 3.1 que ilustra um resumo desses trabalhos. Para cada caso mostrado, existe um resumo alargado abaixo da tabela.

Tabela 3.1: Trabalhos de Junção de Dados pelo Método Não-Ontológico

Nome e Autor do artigo	**Tipo de junção de dados**	**Tipo de Dados**
1 - Pan, Weidong, et al. "An implementation of XML data integration."[PLT08] (2008). (informação retirada de Abstract)	XML	Sem dados definidos.
2 - Savinov, Alexandr. "Concept-oriented model."Encyclopedia of Database Technologies and Applications (2009). [Sav09]	*Concept-Oriented Model* unifica modelos de modelação de dados.	Sem dados definidos.

Continua na próxima página

Tabela 3.1 – continua da página anterior

Nome e Autor do artigo	Tipo de junção de dados	Tipo de Dados
3 - Chromiak, Michał, and Krzysztof Stencel. "A data model for heterogeneous data integration architecture."Beyond Databases, Architectures, and Structures. Springer International Publishing, 2014. 547-556. [CS14]	*Interoperable Data Access Object* (iDAO)	Sem dados definidos.
4 - Salem, Rashed, Omar Boussaïd, and Jérôme Darmont. "Active XML-based Web data integration."Information Systems Frontiers 15.3 (2013): 371-398.[SBD13]	AX-InCoDa	Sem dados definidos.
5 - Key based Approach for Integration of Heterogeneous Data Sources Ahmad, K.; Wook, T.S.F.T.; Samad, R. Source: Journal of Theoretical and Applied Information Technology, v 48, n 2, p 699-703, 20 Feb. 2013 [AWS20]	Junção de Tabelas Relacionais pela Chave Primária	Dados provenientes de várias instituições médicas
6 - Stuckenschmidt, Heiner, Jan Noessner, and Faraz Fallahi. "A Study in User-centric Data Integration."ICEIS (3). 2012. [SNF12]	Centrada no Utilizador com poucos ou nenhuns conhecimentos técnicos.	Sem dados definidos.

1. **Integração com XML**[1] [PLT08] É investigada a implementação de um sistema de integração de dados usando XML. Para isso é efetuada uma junção dos esquemas XML de cada fonte de dados num esquema comum global. É um modelo genérico sem dados-alvo definidos.(informação retirada de Abstract)

2. **Concept-Oriented Model** [Sav09]

[1] Pela impossibilidade de se encontrar o artigo completo disponibilizado nas variadas bases de dados acedidas, decidiu-se fazer a análise superficial baseada somente no *abstract*.

Tal como mencionado neste artigo, *Concept-Oriented Model* é uma abordagem unificada à modelação de dados que generaliza as principais visões sobre dados: relacional, multidimensional, orientada a objeto, conceptual e semântica. É baseada em três princípios estruturais: dualidade, inclusão e ordem parcial.

- **Princípio de Dualidade** indica que um elemento é um par identidate-entidade onde a identidade representa de maneira única a entidade associada.
- **Princípio de Inclusão** indica que todos os elementos existem numa hierarquia onde os elementos-filho são extensões dos elementos-pais.
- **Princípio de Ordem** assume que todos os elementos estão parcialmente ordenados onde elementos menores mencionam a seus elementos maiores.

3. **Interoperable Data Access Object** [CS14]

   Neste artigo é discutida uma solução para efetuar junção de dados heterogéneos tendo por fundamento uma comunicação poliglota baseada num serviço de rede. Esta integra as várias assunções de ambientes de sistemas de gestão das bases de dados, podendo ser adaptada para um eventual novo sistema que se pretende adicionar à integração.

4. **AX-InCoDa** [SBD13]

   No caso de AX-InCoda(*Active XML-based framework for Integrating Complex Data*), devido à complexidade dos dados previstos na utilização, é explorado XML como principal padronização dos dados para um formato unificado, tendo também um papel importante na modelação e armazenamento dos dados. Para integração de dados a partir de fontes de dados distribuídos são usados serviços baseados em meta-dados [ABM08].

5. **Chave Primária** [AWS20]

   No contexto de bases de dados relacionais, é descrito no artigo que tendo em conta que cada tabela relacionada com um paciente possui um ID de paciente. Esse ID varia entre os vários sistemas existentes. O que se propõe é a criação de um identificador único e universal de um paciente[2] que possa ser usado em todos os sistemas.

6. **Centrada no Utilizador** [SNF12]

   Esta abordagem é baseada no modelo de suporte cognitivo que foi implementado na ferramenta de integração de dados *MappingAssistant*. Depois de efetuar um estudo de caso envolvendo pessoas efetuarem junção de dados com esta ferramenta, concluiu-se que esta permite ao utilizador resolver problemas de junção de dados mais eficientemente.

[2]Neste artigo este é designado por IC

3.2 Junção de Dados - Métodos Ontológicos

Nesta secção são analisados os artigos que se acharam mais relevantes para a elaboração deste trabalho. Tendo em conta as ideias ilustradas nestes foram extraídas as mais úteis e aplicadas posteriormente na construção deste sistema.

3.2.1 *Ontology-Based Integration of Data Sources*

Este artigo de Michel Gagnon [Gag07] faz uma revisão dos métodos já existentes de junção de dados através de ontologia. Posteriormente propõe uma integração de informação baseada em mapeamento de ontologias feito a partir das locais para a global, explicando também como é possível a construção de um sistema deste género.

Existem 3 abordagens ontológicas básicas para junção de dados, representadas na Figura 3.1 e originalmente mencionadas em [WVV+01].

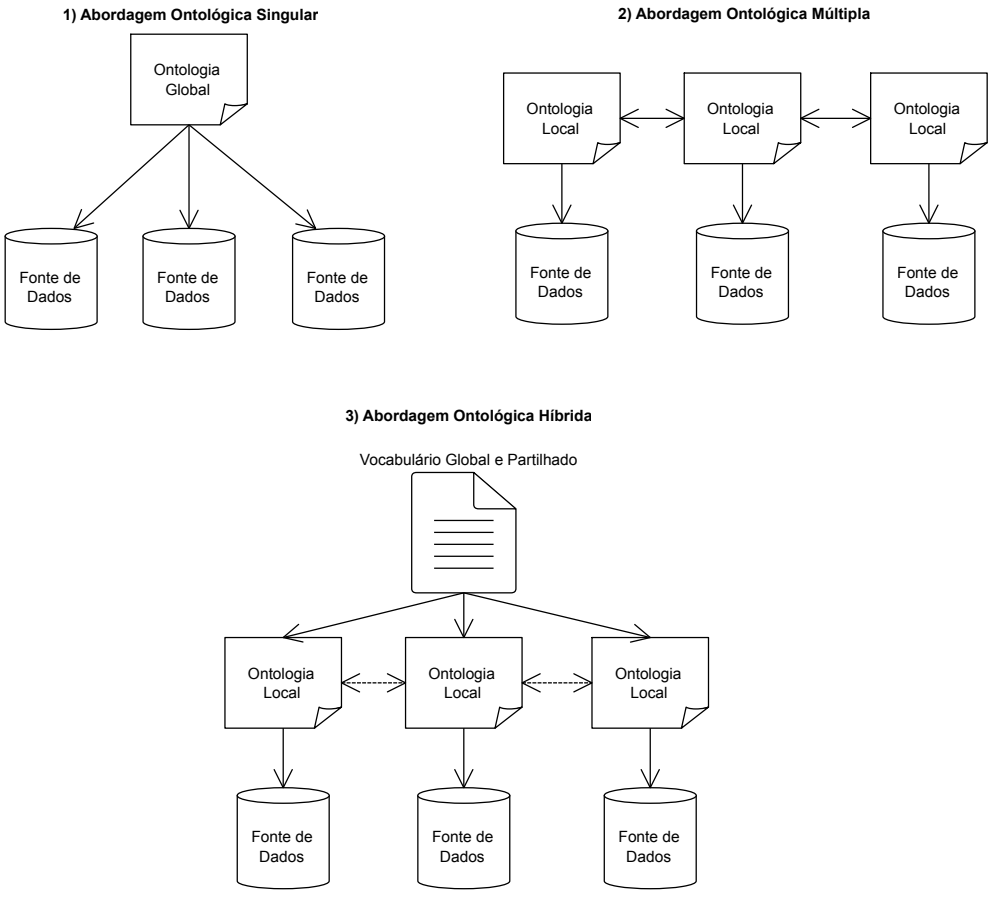

Figura 3.1: 3 Tipos de Abordagens Ontológicas (reproduzido de [WVV+01])

- **Abordagem Ontológica Singular** consiste no uso de uma ontologia global que contém toda a especificação semântica de todas as fontes de dados [WVV+01]. O seu ponto fraco

encontra-se na condição das fontes de dados terem diferentes perspetivas acerca do domínio de discurso. Para isso a próxima abordagem foi elaborada para combater este aspeto [Gru95].

- Na **Abordagem Ontológica Múltipla** cada fonte de dados tem uma ontologia própria para descrever a sua semântica [WVV+01]. Tem imensas vantagens em relação à anterior sendo a principal a inexistência de uma ontologia global que necessita de ser alterada quando uma das fontes de dados sofre alterações estruturais. Por outro lado, a falta de vocabulário comum torna difícil a tarefa de comparação das diferentes ontologias.

- **Abordagem Híbrida** foi desenvolvida para resolver os problemas das duas abordagens anteriores, ao ter caraterísticas provenientes de cada uma das anteriores. Tal como na abordagem múltipla, cada uma das fontes de dados possui uma ontologia que a descreve. Por outro lado para que se possa efetuar comparações entre as diferentes ontologias, estas são construídas a partir de um vocabulário global e partilhado [WSSKR99], que às vezes também é uma ontologia [VSWV00]. Este contém termos básicos do domínio utilizado.

Ao se efetuar a junção de dados, irão certamente surgir conflitos. Michel Gagnon [Gag07] reconhece 3 categorias de conflitos principais:

- **Heterogeneidade Sintática** que diz respeito às diferenças entre os modelos de dados, que podem ser relacionais, orientados a objetos, etc.

- **Heterogeneidade Estrutural ou Esquemática** indicando que cada sistema de informação armazena os seus dados de maneira diferente.

- **Heterogeneidade Semântica** ou diferença de significado do conteúdo de um determinado conjunto de dados. Para que num sistema de informação heterogéneo haja interoperabilidade semântica é condição necessária o entendimento por todas as partes constituintes do significado dos dados que são usados.

Depois de ter analisado os conceitos acima mencionados, de seguida é proposto neste artigo um sistema que consiste na construção de ontologias para cada fonte de informação e também de uma ontologia global que serve de conexão das ontologias locais de cada fonte de dados. Na figura 3.2 é ilustrado o sistema proposto.

Para a construção de um sistema deste género são percorridas 3 etapas:

- **Primeira etapa** consiste na construção das ontologias locais de cada uma das fontes de informação. Isto é necessário para que posteriormente se saiba com maior detalhe informação acerca da estrutura da base de dados local, que na construção da ontologia global pode não ser incluído. Dessa forma, informação implícita acerca do modelo de dados é explícita numa ontologia local.

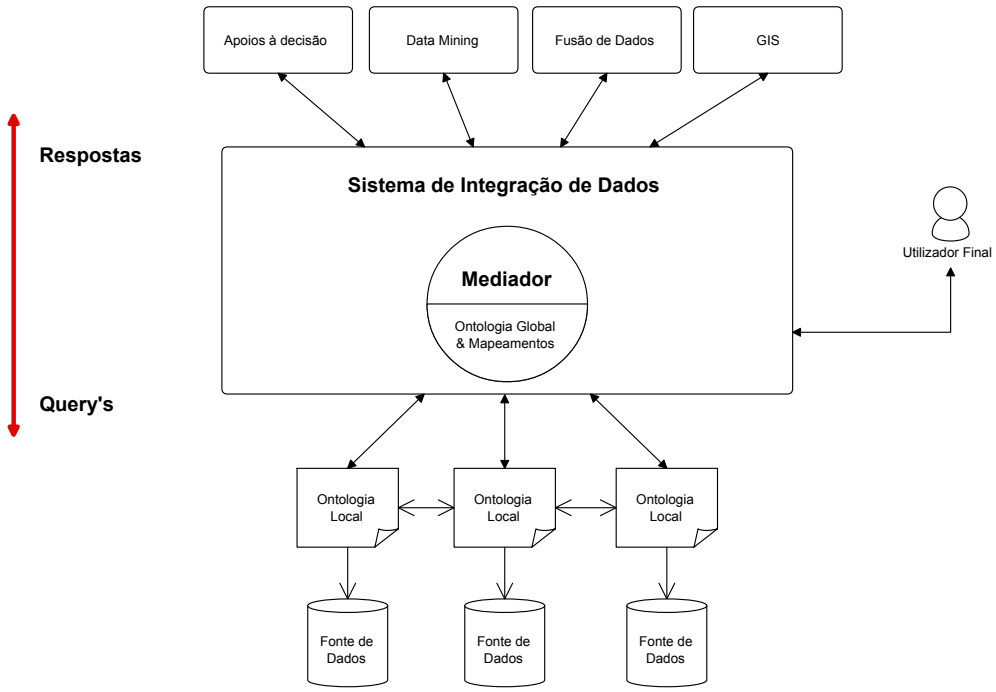

Figura 3.2: Arquitetura de Junção de Dados baseado em Ontologia proposto por [Gag07] (figura reproduzida de [Gag07])

- **Segunda etapa** consiste na construção da ontologia global a partir dos conceitos usados nas ontologias locais.

- **Terceira etapa** é a definição das correspondências entre as fontes de dados, as ontologias locais e a ontologia global. As correspondências são assim guardadas em axiomas e regras do domínio.

Depois de percorridos os passos indicados continua a ser necessário que um especialista do domínio do sistema o valide.

3.2.2 *Quality Aware Service Oriented Ontology Based Data Integration*

Este artigo de Hema e Chandramathi [HC13] propõe uma arquitetura de Integração de Dados Orientada a Serviços com Medição da Qualidade de Serviço (QoS). Esta possui uma camada mediadora que consiste em resolução de conflitos semânticos usando para isso ontologia para criar esquemas locais e global. Possui também outra camada, QoS, que deteta e tenta resolver a imprecisão e a incompletude dos dados provenientes das suas respetivas fontes.

Com isto esta arquitetura é capaz de providenciar ao utilizador dados de alta qualidade. Permite também dar uma visão geral a um utilizador informações sobre os dados presentes no sistema. Notifica também as fontes de dados caso haja imprecisão ou falta de informação nas respostas retornadas.

3.2.2.1 Arquitetura Proposta

A arquitetura proposta está representada na Figura 3.3.

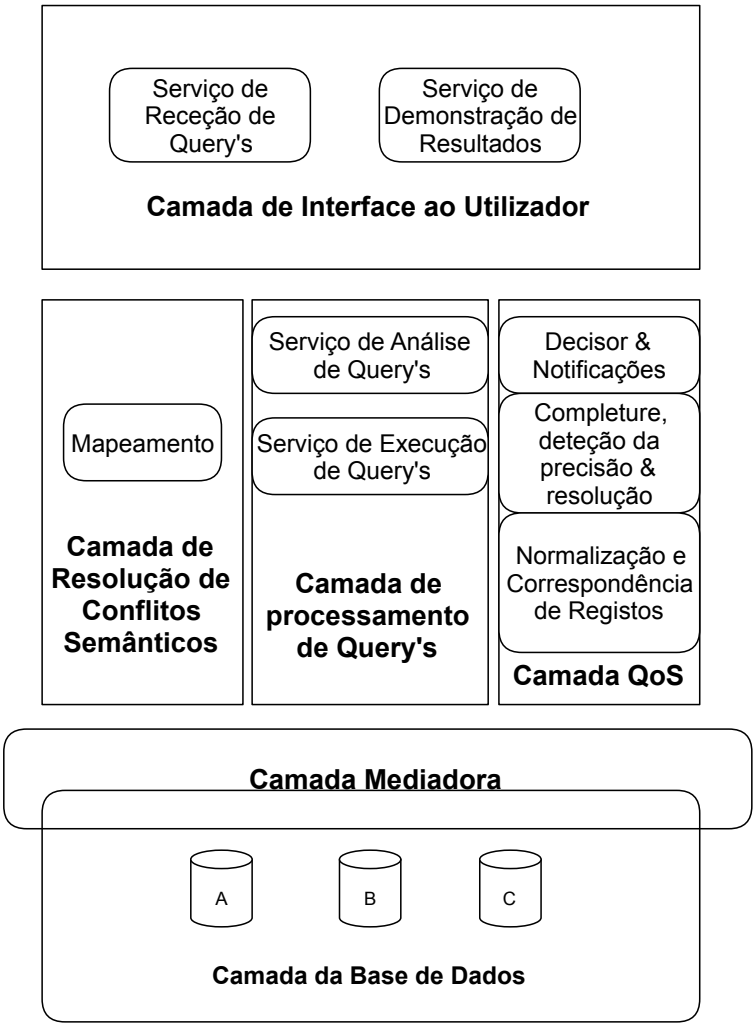

Figura 3.3: SODI-QoS proposto por Hema e Chandramathi (figura reproduzida de)

Esta arquitetura consiste em três camadas com tarefas mutuamente exclusivas.

- A **Camadas das Bases de Dados** contém fontes de dados heterogéneos.
- A **Camada Mediadora** é uma camada intermédia que cria ontologias locais e global e mapeia-as usando ontologia na subcamada de Resolução de Conflitos Semânticos. Esta camada pretende extrair os resultados com qualidade das fontes de dados heterogéneas. A ontologia global foi concebida usando abordagem híbrida.
- A **Camada de processamento de queries** executa as interrogações que um utilizador cria.

- Na **Camada QoS** é detetada e resolvida a incompletude e a imprecisão nos resultados obtidos. Também é a camada que notifica as fontes de informação caso alguma das duas falhas anteriores ocorra.
- A camada superior (**Camada de Interface ao Utilizador**) aceita os pedidos efetuados pelo utilizador.

3.2.2.2 Funcionamento do Sistema

A seguir será explicado em pormenor o funcionamento interligado das várias camadas do sistema proposto.

Na **Camada de Processamento de Queries**, estas são recebidas na linguagem SPARQL e interrogadas à ontologia global. O funcionamento deste processo pode ser definido pelo seguinte algoritmo:

- **Passo 1:** A *query* é interrogada à ontologia global.
- **Passo 2:** A *query* é decomposta em *sub-queries* com base nas regras de mapeamento definidas.
- **Passo 3:** Cada *sub-query* é enviada à respetiva ontologia local.
- **Passo 4:** Usando a camada mediadora, em cada ontologia local a *query* é convertida para a linguagem nativa da base de dados subjacente e enviada à fonte de dados para a obtenção de resultados.
- **Passo 5:** Os resultados são obtidos das respetivas fontes de informação e passadas para a camada QoS para verificação e melhoria da qualidade dos resultados.

A Camada QoS possui várias subcamadas. São elas a **Normalização e Correspondência de Registos**, **Deteção de Incompletude e da Imprecisão** com as respetivas **Resoluções**, e por fim o **Decisor** com a **Notificação** das fontes de informação.

- Na subcamada **Normalização e Correspondência de Registos**, primeiro é efetuada a normalização das bases de dados de maneira a que seja posteriormente possível corresponder campos de uma base de dados com outra.

  Para a segunda parte é utilizado o método probabilístico de correspondência de registos, que usa estimativas sobre se um determinado registo é correspondente a outro. Para isso é usada distância de Jaro-Winkler [CRF03]. Depois de calculada a distância, os registos que representam a mesma informação são fundidos para um *cluster*, ficando esta para a próxima camada a processar.

- Sub-camada de **Deteção de Incompletude e a sua Resolução**. Existem diferentes formas de completude: completude da fonte de dados, completude do tuplo e completude do atributo, sendo que cada uma é calculada de maneira diferente.

$$\textbf{Completude da Fonte de dados} = \frac{NRRS}{TNNR} \tag{3.1}$$

NRRS é o Número de Registos Obtidos a partir da Fonte de Dados e TNNR é o número total de Registos Obtidos.

$$\textbf{Completude do Tuplo} = \frac{NAAT}{TNAR} \tag{3.2}$$

NAAT é o Número de Atributos disponíveis num Tuplo e TNAR é o Número Total de Requisitos Necessários.

$$\textbf{Completude do Tuplo} = \frac{NNNVA}{TNVA} \tag{3.3}$$

NNNVA é o Número de Valores Não-Nulos num Atributo e TNVA é o Número Total de Valores num Atributo.

Os seguintes métodos de resolução são aplicados aos registos resultantes da consulta através da *query* especificada:

- **Resolução 1:** se os valores do atributo dum *cluster* forem exatamente iguais, estes são copiados para o conjunto de resultados sem qualquer modificação.
- **Resolução 2:** se só um ou poucos valores entre os registos comparados tiverem os mesmos valores de atributo dentro de um mesmo *cluster*, é copiado para o conjunto de resultados o registo com o maior valor de completude de tuplo.
- **Resolução 3:** se dois atributos tiverem valores contraditórios mas o mesmo valor de completude de tuplo dentro de um mesmo *cluster*, aí é copiado o registo para o conjunto de resultados com o maior valor de completude da fonte de dados.

O conjunto de resultados é passado para o Decisor e para o Serviço de Notificações.

- Sub-camada **Deteção e Resolução de Imprecisão**. Para esclarecer o contexto, precisão significa a proximidade de um valor *v* a um valor *v'*, considerado correto. Precisão sintática é a proximidade de um valor *v* para com os elementos correspondentes do domínio de definição D. Na medição da precisão sintática o valor *v* não é comparado com o valor *v'* mas sim é verificado se *v* corresponde a algum valor do domínio D.

  Os registos são classificados como **Precisos**, **Imprecisão Fraca** ou **Imprecisão forte** baseadas nas regras da tabela 3.2.

Tabela 3.2: Regras de Previsão de Precisão

Regra Nº	**Parâmetros**	**Previsão**
1	Tuplos Coincidem e são sintaticamente corretos	Preciso
2	Tuplos Coincidem e são sintaticamente incorretos	Imprecisão Fraca
3	Tuplos Não Coincidem e são sintaticamente corretos	Imprecisão Fraca
4	Tuplos Não Coincidem e são sintaticamente incorretos	Imprecisão Forte

Os seguintes métodos de resolução são aplicados, tendo em conta as regras definidas na tabela 3.2:

- **Resolução 1:** se os registos de um *cluster* satisfazem a regra nº1, então é Preciso. Estes são copiados para o conjunto de resultados.
- **Resolução 2:** se os registos do *cluster* satisfazem uma das regras restantes (2, 3, 4), então estes não são precisos. A resolução passa para a parte do Decisor e do Serviço de Notificações.

- E por fim a subcamada do **Decisor Serviço de Notificação**.

  No decisor os valores dos registos são descarregados para o conjunto de resultados. Neste também são tomadas as seguintes decisões:

  - **Decisão 1:** se todos os atributos são completos e precisos então o conjunto de resultados passa para o Serviço de Demonstração de Resultados.
  - **Decisão 2:** se existe alguma incompletude ou imprecisão no conjunto de resultados então o conjunto de resultados é na mesma passado para o Serviço de Demonstração de Resultados e também para o Serviço de Notificação.

  O serviço de notificação assegura-se de notificar a fonte de dados caso a informação desta seja incompleta ou imprecisa.

3.2.3 *A Fuzzy Ontology-based Semantic Data Integration System*

Este artigo de Cristiane A. Yaguinuma et al. [YAF+11] menciona uma junção efetuada que é diferente, tendo em comparação os artigos já analisados. Esta consiste no mapeamento das informações existentes nas fontes de dados com informação geográfica.

Para isso foi desenvolvido o sistema DISFOQuE, um sistema de integração de dados que usa ontologias difusas para integrar dados heterogéneos de variadas fontes de informação. Para além disso, utiliza conceitos e relações difusos para executar a expansão semântica das *queries* para obter resultados aproximados que possam ser relevantes ao utilizador.

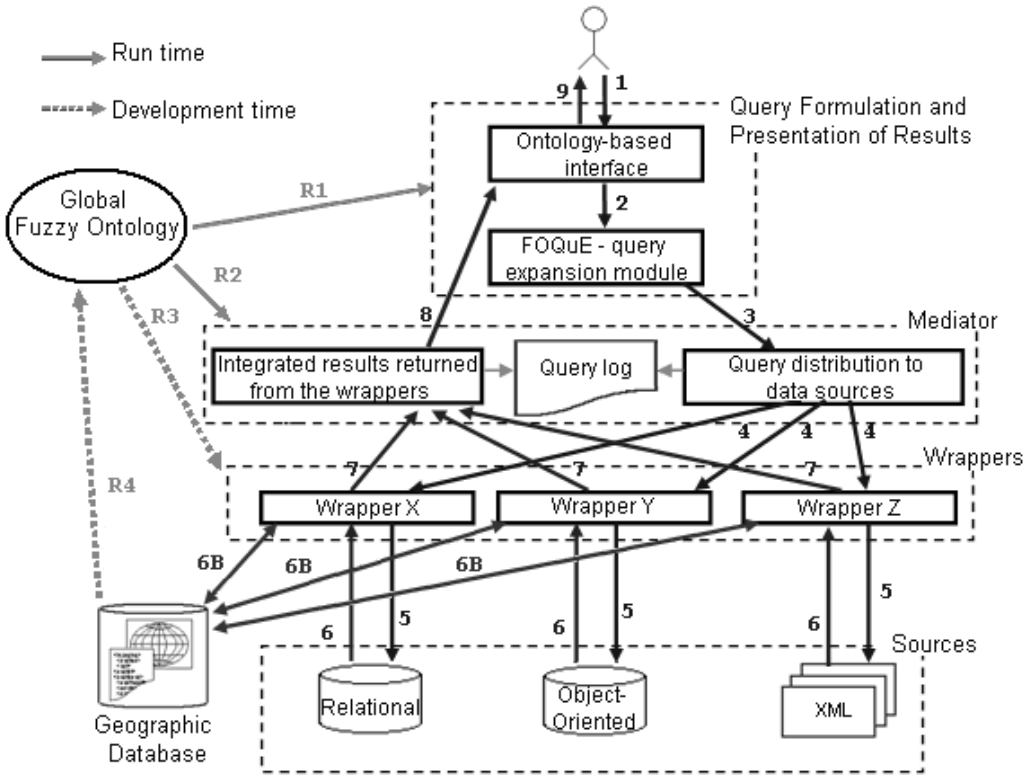

Figura 3.4: Arquitetura do sistema DISFOQuE (reproduzido de [YAF+11])

A arquitetura deste sistema está ilustrado na Figura 3.4.

Em termos gerais este sistema é bastante parecido com o artigo analisado anteriormente na Secção 3.2.2 em termos de arquitetura, mas a parte importante que será aproveitada neste trabalho é a ligação da Camada Mediadora, que interage entre as fontes de dados e as camadas superiores, que apoiando-se numa base de dados geográfica (setas 6B na Figura 3.4) verifica se aquando de uma resposta por parte das fontes de dados, existe ou não informação relacionada com dados geográficos. Isto permite obter resultados corretos e relacionados com localização espacial [YAF+11].

3.2.4 *An extended hybrid ontology approach to data integration*

Neste artigo, Zhang Laomo, Ying Ma e Guodong Wang [ZMW09] propõe uma extensão ao método ontológico híbrido, que segundo eles possui limitações. Estas limitações baseiam-se em que as ontologias já existentes são difíceis de reutilizar e que têm que ser refeitas.

Para resolver o problema [ZMW09] propõem que para cada fonte de dados seja criado um esquema local em XML sobre o qual será construída a ontologia. Na Figura 3.5 é apresentado o esquema da abordagem híbrida estendida.

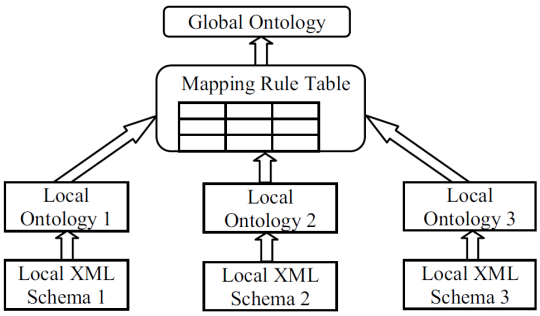

Figura 3.5: Abordagem Ontológica Híbrida Estendida (imagem reproduzida de [ZMW09])

3.2.5 *Relational Schema Integration Using Ontologies*

Na dissertação [Pan14] o tema principal é a integração de Modelos Relacionais usando Ontologias. Para isso, o autor primeiro aborda a conversão dos esquemas relacionais para as ontologias e posteriormente aplica métodos de junção de ontologias para obter uma ontologia final. Depois disso essa ontologia final é convertida de novo para um esquema relacional.

3.3 Ferramentas Relacionadas

Nesta secção irão ser descritas de uma maneira breve as ferramentas que foram encontradas e que de alguma forma se relacionam com o trabalho desenvolvido. Esta secção também irá ter uma descrição pormenorizada com as diferenças entre as ferramentas já existentes e a solução que é elaborada nesta dissertação, discutida na Secção 4.2. Uma breve comparação entre as ferramentas pode ser visualizada na Tabela 3.4.

3.3.1 *Ontology-based geospatial data query and integration*

Neste artigo é descrita uma ferramenta que tem por base uma ontologia. Esta ontologia foi criada com o propósito de descrever um sistema de rotas de autocarros. Para aceder a dados alocados em bases de dados e WFS(*Web Feature Service*), é feito um mapeamento designado de vistas RDF (RDF views). Este processo de mapeamento e de ligação da ontologia à base de dados ou WFS é feito manualmente. Com estas vistas RDF o processo de tradução das *queries* SPARQL para *SQL* ou *getFeature* é simplificado [ZZWP08].

As principais diferenças entre a ferramenta descrita neste artigo e Trontegro estão representadas na Tabela 3.3

3.3.2 *Silk - The Linked Data Integration Framework*

Silk[3] é uma *framework* que é usada para integrar variadas fontes de dados heterogéneas. Na sua essência esta ferramenta é baseada nos conceitos de *Linked Data*[VBGK09]. Silk pode ser

[3]http://silk-framework.com/

Tabela 3.3: Diferenças entre [ZZWP08] e Trontegra

Caraterística	**[ZZWP08]**	**Trontegra**
Ontologia Geral	Construída manualmente	Construída automaticamente
Mapeamento	É feito por vistas RDF que mapeiam a ontologia conforme a fonte de dados. Método manual.	Método de mapeamento automático usando d2rq
Tipo de Dados	Somente dados que de alguma forma se relacionem com a ontologia	(hipoteticamente) Quaisquer dados, testados só espacio-temporais
Tipos de Fontes de Dados	Bases de dados relacionais e WFS	csv, osm.pbf e bases de dados relacionais

usada por desenvolvedores para especificar regras de acordo com as quais irão ser descobertas possíveis ligações RDF entre fontes de dados.

Em comparação ao Trontegra, Silk estabelece ligações RDF entre os dados obtidos a partir de armazéns de dados RDF e não permite a adição de fontes de dados na forma de bases de dados relacionais. Sendo o acesso aos dados feito diretamente por RDF, não é feita tradução de SPARQL para SQL. No entanto e tal como Trontegra, Silk interroga as fontes de dados através dos seus pontos de acesso SPARQL de cada uma das fontes de dados.

3.3.3 Datalift

Datalift[4] é uma plataforma que converte múltiplos formatos de fontes de dados em *Linked Data*. O formato das fontes de dados varia desde bases de dados até ficheiros XML ou CSV, entre outros [SAT+12].

Em termos de fontes de dados suportadas, os sistemas Datalift e Trontegra são parecidos, mas os seus métodos de adição e de integração das fontes de dados são diferentes. Esta diferença é:

- Em Trontegra ficheiros CSV e OSM são importados para bases de dados MySQL e PostgreSQL respetivamente.

- Em DataLift é feita uma referência para o ficheiro sem o processar. Para aceder à informação do ficheiro, este tem de ser completamente convertido para RDF, sendo esta a diferença chave entre as duas aproximações.

Um teste comparativo foi feito, em que se usou um ficheiro CSV de 100MB. Em Trontegra o processo demorou 3 minutos a terminar e o espaço em disco ocupado pelo processo foi de 80MB (remoção de campos repetidos foi automaticamente feito pelo MySQL). Em Datalift esta conversão para RDF demorou perto de 35 minutos e ocupou em disco um total de 3GB.

[4] http://datalift.org/

3.3.4 *The Mastro System for Ontology-based Data Access*

Mastro[5] é uma ferramenta que permite aceder a dados provenientes de variadas fontes de dados através de uma ontologia geral [CDGL+11].

Existem semelhanças entre os sistemas Mastro e Trontegra:

- cada um deles possui um mecanismo de obtenção do esquema das bases de dados.
- cada um deles tem um tradutor de SPARQL para SQL e vice-versa.
- cada um deles tem um ponto de acesso SPARQL que permite aceder a várias fontes de dados presentes no sistema.

Em termos gerais Trontegra é uma ferramenta parecida com Mastro mas tem já implementado um serviço de visualização de dados e tem aplicação direta nas fontes de dados espaciotemporais. Será de considerar a hipótese de Trontegra ser baseado em Mastro, mas este último foi somente encontrado nos estágios finais de desenvolvimento de Trontegra.

3.3.5 GeoTriples

GeoTriples[6] é uma ferramenta que permite a um utilizador publicar dados geoespaciais como *Linked Data* [KVS+]. Apesar desta não criar uma ontologia geral nem permitir ao utilizador integrar variadas fontes de dados, aplica um modelo ontológico em dados geoespaciais.

3.4 Sumário

Este capítulo permite ver algumas abordagens de junção de dados. Estas podem tanto ser feitas por abordagens puramente ontológicas mas também utilizando outros métodos. No entanto, sendo este trabalho dirigido para a abordagem ontológica, foi dado um foco maior nessa parte da literatura.

Depois de estudar as abordagens [Gag07] e [HC13], obtiveram-se as bases para a construção de Trontegra. Uma ontologia local será criada para cada uma das fontes de dados e posteriormente uma ontologia global será construída a partir das várias locais. Ou seja, a metodologia usada é a Híbrida.

A partir da análise à literatura, também foi possível perceber que não há muitas abordagens ontológicas em dados espaciotemporais. Daí Trontegra já ser uma contribuição científica nesse aspeto.

[5] http://www.dis.uniroma1.it/ mastro/
[6] https://github.com/LinkedEOData/GeoTriples

Tabela 3.4: Comparação entre as ferramentas encontradas e Trontegra

Ferramenta / Caraterística	[ZZWP08]	**Silk**	**Datalift**	**Mastro**	**GeoTriples**	**Trontegra**
Tipos de Fontes de Dados	Bases de dados e WFS	Armazém de dados RDF	CSV, XML, RDB, RDF, SPARQL endpoint, Shapefile, GML	RDB	Dados Geoespaciais	CSV, OSM e Bases de Dados Relacionais
Integração de fontes de dados	Vistas RDF	Ligações RDF	Ligações RDF	Desconhecido	Não	Ontologia global criada a partir da junção das ontologias locais
Serviço de Interrogação SPARQL	Sim	Sim	Sim	Sim	Não	Sim
Interrogação de várias fontes de dados	Sim	Não	Sim	Sim	Não	Sim
Serviço de Visualização de Dados	Sim	Não	Não	Não	Não	Sim
Aplicação em dados espaciotemporais	Sim	Não	Geoespaciais	Não	Geoespaciais	Sim

Capítulo 4

Solução Proposta - Trontegra

Neste capítulo é apresentada uma revisão do problema que levou à construção do Trontegra, que visa resolver o problema identificado e nos seus casos de utilização práticos. No caso da solução, são discutidas tanto a abordagem genérica que inclui todas as funcionalidades pretendidas para a resolução completa do problema e a abordagem prática também designada de prova de conceito, que é a implementação da solução genérica. Esta serve para tentar comprovar a eficácia do conceito genérico na resolução do problema proposto.

4.1 O Problema

O problema que se pretende resolver é a grande dispersão de dados e fontes de dados espacio-temporais existente. Estes dados tanto podem estar armazenados numa base de dados como num ficheiro **csv**. Esta vastidão requer que para cada fonte de dados exista um ponto de acesso único e com uma estrutura própria. Daí, posteriormente a análise dos dados se torna mais difícil por causa desta dispersão.

Embora já existam formas de acesso a estas fontes de dados, isto não facilita de forma alguma a utilização simultânea de dados provenientes de locais diferentes. A heterogeneidade presente implica que um utilizador que pretenda utilizar simultaneamente dados de p.ex. *espiras magnéticas* e *logs GPS* tenha de implementar uma interface capaz de interrogar ao mesmo tempo estas duas fontes de dados. Isto demora tempo de implementação, de teste e de adaptação ao novo sistema. A eventual posterior adição de uma nova fonte de dados ao sistema implicará novamente um esforço de implementação e teste da ferramenta. São estas as caraterísticas do problema presente que forçam a investigação de novas formas de integração das várias fontes de dados num só sistema.

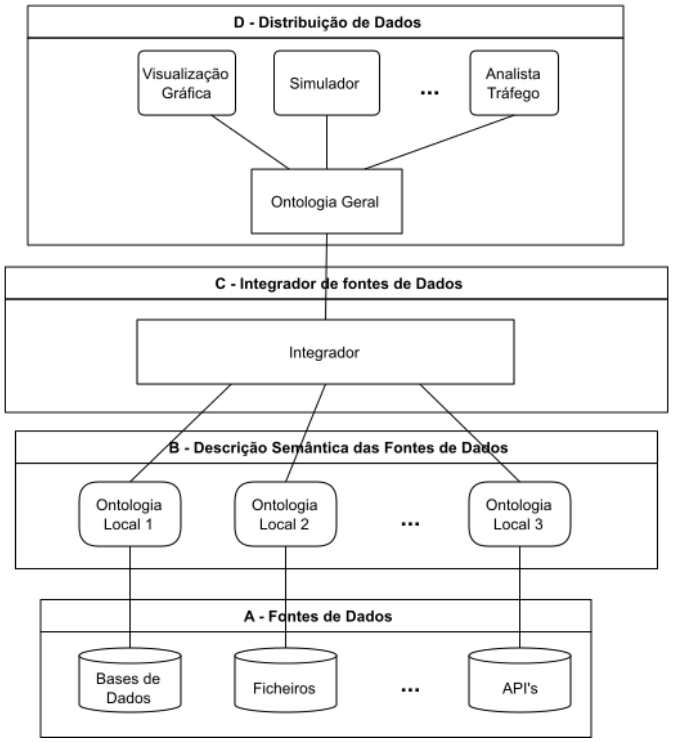

Figura 4.1: Arquitetura da Solução Genérica Proposta

4.2 Visão da Solução Genérica

Como solução genérica para este problema, pretende-se uma abordagem ontológica. Com este modelo os dados têm significado bem definido e que pode ser percebido por agentes automáticos. Ao aplicar esta abordagem também é possível juntar as variadas fontes de dados só num modelo ontológico. Isto vai permitir com que todos os dados presentes no sistema possam ser acedidos de um só ponto.

Para uma melhor compreensão da solução genérica, foram elaboradas duas figuras. Na Figura 4.1 está representada a sua arquitetura e na Figura 4.2 está representado um diagrama de atividades.

Passando a explicar os conceitos mais profundos da solução genérica proposta, pode-se considera que o sistema tem 4 partes principais, e são elas:

- Camada A - Fontes de Dados
- Camada B - Descrição Semântica das Fontes de Dados
- Camada C - Integrador de Fontes de Dados
- Camada D - Distribuição de Dados

A camada A é responsável pelo conjunto das fontes de dados presentes no sistema, sendo também o ponto inicial da execução deste. É aqui que será possível adicionar de maneira simplificada uma nova fonte de informação. Esta é analisada e os seus identificadores (classes, colunas, etc.)

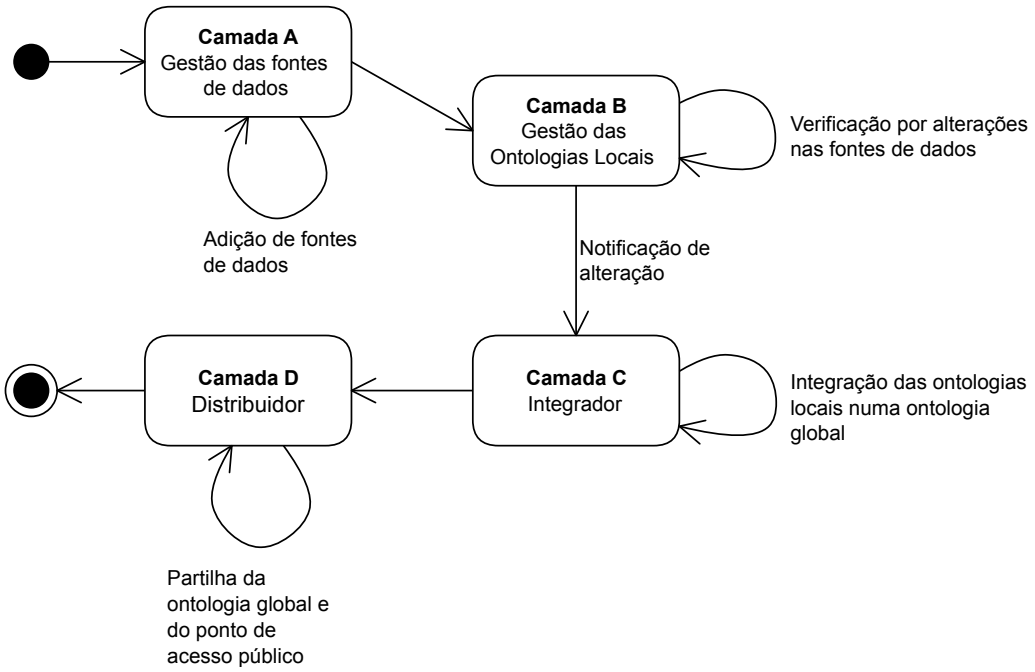

Figura 4.2: Diagrama de Atividades da Solução Genérica

mapeados para uma ontologia pertencente já à camada B. Podem ser aceites pelo sistema variados tipos de formatos de fontes de dados. São exemplos disso as bases de dados, ficheiros XML e variantes (KMZ, GPX), ficheiros CSV e variantes, API's, pontos de acesso SPARQL, *output* de resultados de simuladores, ficheiros OSM, e por fim outras ontologias que possam de alguma forma posteriormente trazer alguma mais valia aos dados ou às fontes de dados.

Na camada B são armazenadas as ontologias locais pertencentes às fontes de dados presentes na camada A. Cada uma destas ontologias, designadas de locais, tanto servirá como um facilitador de acesso aos dados como uma descrição formal do modelo dos dados praticado pela fonte correspondente, e que posteriormente será integrada na ontologia global. A criação deste modelo ontológico é feito de forma automática para cada tipo de fonte de dados. Este poderá também ser facilmente atualizado, alterado ou até removido sem que o sistema interrompa o seu correto funcionamento. Ao ser efetuada alguma das alterações mencionadas anteriormente, a camada C é notificada de que houve uma alteração nas ontologias locais, tomando para isso as ações necessárias.

Depois de ter um bom conjunto de fontes de dados que já possuem mapeamento para um modelo ontológico, é necessário integrá-las numa só representação genérica. Isto é da responsabilidade da secção C. As suas tarefas principais incluem processar todas as ontologias presentes na camada B e integrá-las numa única ontologia e atualizá-la caso haja alguma modificação proveniente da camada B. Este processo de integração usa variadas formas de correspondência entre as muitas classes ou indivíduos já existentes nas ontologias locais, de forma a encontrar conexões entre as variadas fontes de dados e para permitir a execução de interrogações globais mais eficientes.

A parte da interrogação é executada na camada D.

O ponto de acesso público a todos os dados do sistema é gerido pela camada D. Esta é também responsável pela publicação da ontologia geral, que permite aos serviços que pretendam dar uso aos dados existentes no sistema saber como estes estão organizados. Muitos são os serviços que podem utilizar os dados, e o poderão fazer de forma simultânea. Podem ser eles uma simples visualização gráfica, reutilização destes por um simulador ou até por um analista de tráfego. Os dados poderão ser acedidos de variadas formas, tanto por exportação em variados formatos como por interrogações a um ponto de acesso SPARQL. Cada interrogação que é recebida pela camada D é enviada para a C, que a processa e divide em interrogações locais, que são enviadas para a camada B para as ontologias locais correspondentes. Estas por sua vez convertem-nas para o formato de interrogação aceite pela fonte de dados, e enviam para estas esperando pela resposta. O processo reverso é aplicado quando as fontes locais da camada A respondem, sendo as respostas processadas e convertidas para SPARQL pela camada B, agregadas pela camada C e enviadas de volta ao serviço pela camada D.

4.3 Casos de Utilização

Por ser um sistema que agrega muitas fontes de dados simultaneamente, este torna-se interessante para serviços que queiram consumir esses dados. Mais nomeadamente serviços que de alguma forma estão relacionados com dados espaciotemporais ou até geográficos.

Consideremos o caso em [MPS+14] onde dados provenientes de variadas fontes providenciam caminhos personalizados para pessoas com mobilidade reduzida. Neste caso o sistema proposto poderá servir dados para este serviço pois possui a caraterísticas de suporte de múltiplas fontes, as quais poderão incluir tanto a informação relativa ao mapa que se apresenta ao utilizador, como também as caraterísticas da topologia de cada caminho tem. Desta forma efetuando uma interrogação geral será possível saber para um determinado ponto no mapa, toda a informação que as outras fontes de dados possam eventualmente ter relativamente àquela posição.

Outro serviço possível é o de representação visual dos dados presentes no sistema. Neste caso, tanto os dados do mapa como os que o povoam (marcadores, linhas, descrições, etc) podem ser obtidos a partir deste sistema. Isto poderá trazer benefícios aos analistas de trânsito para identificar possíveis melhorias que se possam fazer às estradas já existentes ou até para criação de caminhos alternativos.

Um simulador também poderá gozar dos benefícios deste sistema. Isto porque existindo muitas fontes de dados heterogéneas, poderão daí ser extraídas variáveis que de alguma forma favorecerão a simulação e fazem com que os resultados obtidos sejam mais precisos, em diferentes resoluções [PRK11].

Adicionalmente, e fazendo uso de ambientes de simulação, a implementação de jogos baseados na localização também poderá beneficiar de uma plataforma como o Trontegra. Neste contexto, jogos sérios têm sido utilizados no domínio dos transportes [RAKG13], como a modelação comportamental de motoristas [RL05, AGR+13, OMR14] e a avaliação da ergonomia de sistemas

de informação em veículos [GGROM14, GJR+15, GROM12, GRJ+14]. Adicionalmente, em ambientes urbanos estudos têm sugerido a utilização de jogos na avaliação de planos de emergência e evacuação de edifícios [ARFC15, RAR+12]. As informações necessárias à configuração de tais cenários podem ter origem em diferentes sensores e fontes de dados abertas.

4.4 Prova de Conceito - Abordagem Escolhida

Depois de estudar a solução geral proposta, e tendo em atenção os limites de tempo estabelecidos, foram acordadas as caraterísticas da prova de conceito. Estas servirão para tentar comprovar a funcionalidade do sistema genérico proposto. As caraterísticas são apresentadas a seguir.

Foi também decidido dar um nome ao sistema a criar: Trontegra. Este nome provém de três palavras relacionadas com o trabalho **Tr**ansportes, **On**tologia e In**tegra**ção.

4.4.1 Adição de Novas Fontes de Dados

Esta funcionalidade será limitada para ficheiros CSV, OSM e bases de dados relacionais, sendo que os ficheiros CSV e OSM são importados para uma base de dados relacional, de forma a terem um tratamento destes mais eficiente e também para gozarem de todas as caraterísticas de a informação numa base de dados ser indexada. A própria ação é facilitada utilizando ferramentas específicas já embebidas no Trontegra. Neste processo é feito um mapeamento do esquema de dados para uma ontologia onde esta descreve as tabelas e os campos existentes na base de dados. Este mapeamento fará com que ao iniciar Trontegra será também estabelecido um ponto de acesso SPARQL local para cada uma das fontes de dados. Estes pontos de acesso receberão as *queries* locais enviadas pelo ponto de acesso SPARQL geral.

Esta abordagem tem algumas limitações, como o facto do mapeamento dos campos das tabelas incluir o tipo de dados que cada campo tem, mas não ser adicionada mais nenhuma semântica em relação a isso. No caso dos ficheiros CSV também é possível escolher os tipos de dados que cada coluna tem, antes de importar este ficheiro para a base de dados, não sendo também nenhuma informação semântica adicionada posteriormente.

4.4.2 Integração

A partir das ontologias locais é extraída a informação relativa às tabelas e campos. Posteriormente estes dados são adicionados como indivíduos de classes duma ontologia geral já criada previamente. Posteriormente esta será publicada como a ontologia geral e é com base nesta que *queries* gerais poderão ser feitas.

Em comparação com a solução genérica, caso alguma das fontes de dados seja alterada ou removida, o mapeamento terá de ser refeito ou apagado no segundo caso. Depois desta operação feita, é necessário reiniciar a ferramenta para que esta volte a funcionar.

4.4.3 Ponto de Acesso Geral SPARQL

Para que seja possível interrogar várias fontes de dados ao mesmo tempo utilizando a ontologia geral, será necessário manualmente construir as *queries*. Estas precisam de incluir a informação relativa ao nome dos campos e da(s) fonte(s) de dados que se quer interrogar e enviá-la para o ponto de acesso geral SPARQL. Esta *query* é processada e dividida em *queries* locais para ser enviada aos pontos de acesso correspondentes.

4.4.4 Serviço de Visualização

Para facilitar a escrita das *queries* e para visualizar os dados existentes no sistema, criou-se um visualizador. Este apresenta um mapa e os dados são carregados a partir do ponto de acesso geral SPARQL.

4.5 Sumário

Neste capítulo foi abordado a solução genérica, seguida da abordagem escolhida para implementar a solução do problema proposto. Foram também indicadas quais são as limitações dessa abordagem e como em termos gerais esta funciona. Para o sistema proposto foram também apresentados possíveis casos de utilização.

Em termos desta dissertação, este é o capítulo que explicou toda a informação relativa à solução genérica e da prova de conceito, que são ambas constituídas por 4 módulos principais que interagem entre si. Em termos de fontes de dados suportadas, decidiu-se ficar pelos ficheiros CSV e OSM e por bases de dados relacionais. Estes são mapeados para as respetivas ontologias locais que posteriormente são integradas na ontologia geral ou genérica. A partir desta, é aberto um ponto de acesso SPARQL que permitirá aos serviços interrogarem os dados presentes no sistema.

Capítulo 5

Implementação e Análise de Resultados

Neste capítulo irá ser explicada inicialmente a arquitetura do sistema desenvolvido. Também será relatado em que ambiente foi desenvolvido o sistema e onde este poderá ser corretamente reproduzido. Serão também mencionados as ferramentas já existentes que foram incorporadas no Trontegra. Posteriormente serão abordados os problemas encontrados durante a implementação e as falhas que se encontrou ao utilizar as ferramentas externas. No final deste capítulo serão apresentados os testes que foram efetuados à ferramenta que serviram para a sua avaliação. Com base nos resultados dos testes será também feita uma extensa discussão acerca da validade, viabilidade e da eficiência da ferramenta.

5.1 Ferramentas usadas

Nesta secção são brevemente mencionadas quais as ferramentas que foram utilizadas no Trontegra. Todas estas estão embutidas no sistema para que o utilizador não tenha que se preocupar com a maneira de as executar.

5.1.1 d2rq

A ferramenta d2rq[1] tem várias responsabilidades no sistema Trontegra. É usada para gerir a ligação entre cada base de dados e os pontos de acesso SPARQL locais. Também é a responsável pela geração de mapeamentos do esquema relacional da base de dados para a ontologia local.

5.1.2 Apache Jena

Apache Jena[2] permite ao sistema ler as ontologias locais e a ontologia-base que serve de molde à ontologia geral que será posteriormente criada. Depois de efetuado o processamento das

[1] http://d2rq.org/
[2] https://jena.apache.org/

mesmas, que não é feito pela Apache Jena, permite também escrever a ontologia geral final para um ficheiro.

5.1.3 Apache Jena Fuseki

Apache Jena Fuseki[3] é responsável por carregar a ontologia geral criada em memória. Também é responsável por publicar essa mesma ontologia para que eventuais clientes do sistema possam explorar a informação existente mais facilmente. Serve também de ponto de acesso geral de acesso aos dados do sistema.

5.1.4 OpenCSV

OpenCSV[4] é um *parser* utilizado para ler os ficheiros **csv** que se queiram importar para o sistema.

5.1.5 Osm2pgsql

Osm2pgsql[5] é a ferramenta que é utilizada para importar ficheiros **osm.pbf** para uma base de dados local PostgreSQL. Este tipo de ficheiros contém informação vetorial proveniente de **OpenStreetMaps**[6] relativa à topologia das estradas. Esta informação é utilizada para desenhar as estradas no serviço de visualização explicado na secção 5.2.5.

5.1.6 Leaflet

Leaflet[7] é uma biblioteca escrita na linguagem *Javascript*. Esta é utilizada pelo serviço de visualização para mostrar a informação espaciotemporal existente no sistema.

5.2 Arquitetura e Implementação da Plataforma

Na Figura 5.1 é apresentada de forma completa a arquitetura do sistema construído. A seguir cada uma das partes constituintes irá ser explicada com mais pormenor.

5.2.1 Adição de Fontes de Dados

Na Figura 5.2 é demonstrada a parte da arquitetura onde estão representadas as fontes de dados. Trontegra suporta três tipos: ficheiros csv, osm.pbf e bases de dados relacionais. Quanto às últimas, são somente suportados MySQL e PostgreSQL. A sua adição ao sistema é possível por meio de um formulário que Trontegra apresenta ao utilizador.

[3] https://jena.apache.org/documentation/fuseki2/index.html
[4] http://opencsv.sourceforge.net/
[5] https://wiki.openstreetmap.org/wiki/Osm2pgsql
[6] https://www.openstreetmap.org/
[7] http://leafletjs.com/

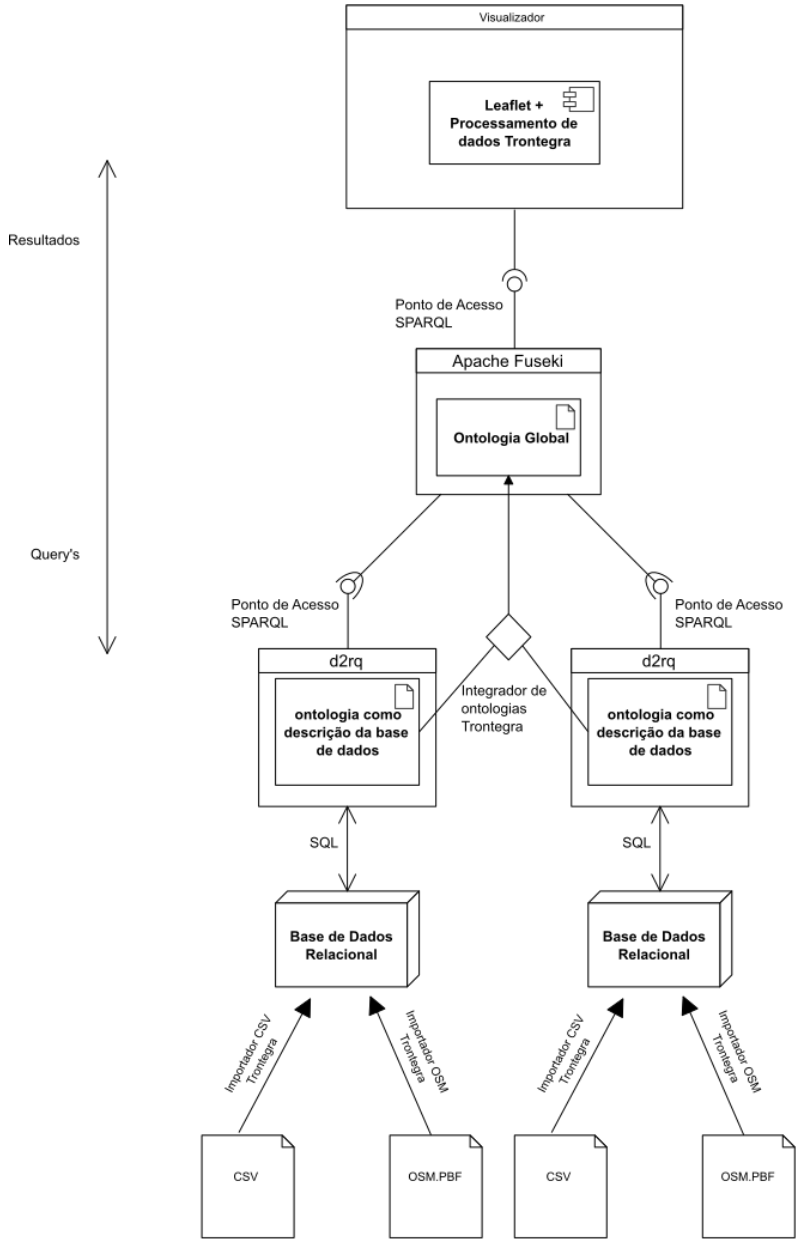

Figura 5.1: Arquitetura do Sistema

A implementação deste módulo foi feita em Java. Para a importação de cada uma das fontes de dados foi feita uma interface bastante simples e que em seguida se passa a descrever:

- Base de dados relacional - Aquando de adição de uma fonte de dados deste tipo, é utilizada a ferramenta d2rq já embutida no Trontegra. Esta é responsável pela comunicação com a base de dados, leitura do esquema relacional e do seu mapeamento para a ontologia local. Esta ferramenta sendo uma biblioteca que já vem pré-compilada para um ficheiro binário, é inicializada através da criação de um novo processo. Depois desta terminar o processo de mapeamento, é criado um ficheiro *.ttl* na pasta que guarda todos os mapeamentos já feitos.

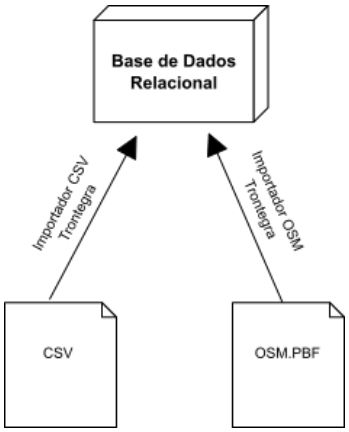

Figura 5.2: Adição de Fontes de Dados

- csv - No caso dos ficheiros CSV, durante a importação destes o utilizador tem a hipótese de selecionar para cada uma das colunas de dados do ficheiro o tipo de dados correspondente. O utilizador tem também a possibilidade de escolher qual das colunas é a chave primária, ou caso não saiba criar uma nova. Toda esta informação é utilizada na *query* SQL usada para a criação da base de dados MySQL para aquele ficheiro em específico. Estes ficheiros são lidos linha a linha usando a biblioteca OpenCSV.

  Os dados são de seguida importados para a base de dados criada anteriormente. À medida que o ficheiro é lido são criadas *queries* de inserção de dados com blocos de 100 linhas por *query* para acelerar o processo de importação. Depois de este procedimento terminar, é aplicado o mesmo que o da importação de uma base de dados relacional.

- osm.pbf - Estes ficheiros contém a informação proveniente de OpenStreetMaps que descrevem a topologia das estradas de uma determinada zona geográfica. Em primeiro lugar este é importado para uma base de dados relacional, mas neste caso PostgreSQL. Para que isso seja feita com sucesso, é utilizada a ferramenta Osm2pgsql que trata de todo o processo de conexão e de importação destes dados. Quando este termina, é aplicado o mesmo procedimento que o da importação de uma base de dados relacional, embora depois de o processo estar concluído, o ficheiro .ttl gerado é alterado de maneira a que a integração e interrogação desta fonte de dados seja facilitada e mais completa.

Em termos de problemas encontrados, inicialmente pretendia-se incluir ainda mais formatos de fontes de dados como xml, kml e gpx. Mas por causa da falta de ferramentas de conversão destes formatos para ontologia estes não foram incluídos. A implementação dos conversores não era um possibilidade devido à limitação do tempo e também porque não faz parte do tópico principal deste trabalho.

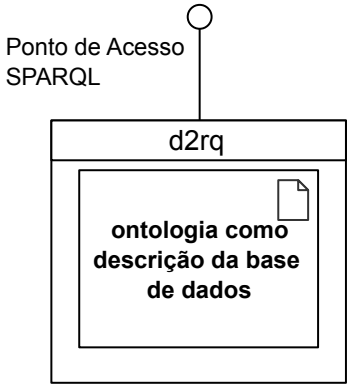

Figura 5.3: Servidor d2rq com ontologia local

5.2.2 Ontologias e Pontos de Acesso SPARQL Locais

De maneira a que seja possível interrogar as fontes de dados presentes no sistema, estas necessitam de ter um ponto de acesso SPARQL ativo. Para tal, é igualmente utilizado o d2rq para inicializar um servidor SPARQL, designado de *d2rq-server*. Cada um destes servidores permite ter tanto um ponto de acesso SPARQL ativo como também publicada a ontologia local de cada fonte de dados. Todas as instâncias dos servidores d2rq são controladas pelo Trontegra.

De forma a que este processo seja automático, ao inicializar as fontes de dados locais, ou seja, ao iniciar os servidores d2rq relativos a cada um dos mapeamentos feitos, é criada em Java uma *thread* que executa o ficheiro binário correspondente, criando por isso um novo processo. Esta *thread* lê a saída que cada um processos produzem, reproduzindo isso para uma consola de forma a que o utilizador tenha o *feedback* de como está a correr o processo de inicialização. À medida que os processos iniciam, são também adicionados a um ficheiro de controlo os respetivos números de processo de cada um dos servidores juntamente com o nome do ficheiro de mapeamento utilizado. Isto terá duas aplicações práticas: é usado para terminar os processos caso Trontegra seja encerrado e é utilizado para efeitos de verificação no ato de integrar as fontes de dados.

Depois de inicializado, é possível aceder a cada uma das fontes de dados através do url http://localhost:[porta], em que [porta] é a porta correspondente à fonte de dados que se quer aceder. O ponto de acesso SPARQL está disponível em http://localhost:[porta]/sparql.

Em termos de problemas encontrados, a maior parte destes são relacionados com a ferramenta d2rq. Por exemplo quando é feita uma interrogação em SPARQL que inclui filtragem de strings por meio do comando **REGEX**, este não é traduzido para SQL. Por isso o que d2rq faz é converter a *query* SPARQL para SQL sem a parte de filtragem por regex e a aplica depois de receber a resposta. Isto causa um acréscimo grande no tempo de execução do processo de interrogação.

O suporte de d2rq para a versão SPARQL mais recente está limitada pelo uso de Joseki [8], uma biblioteca que implementa um servidor SPARQL utilizado no interior da ferramenta d2rq. Joseki neste momento é obsoleto e o seu desenvolvimento parou, pois este foi substituído pelo Fuseki [9]

[8] http://sourceforge.net/projects/joseki/
[9] https://jena.apache.org/documentation/fuseki2/index.html

que tem exatamente o mesmo propósito. Apesar de ter acontecido esta troca, essa transição não ocorreu em d2rq, pelo que por exemplo o suporte a interrogações de bases de dados relacionais com muitas entradas é atualmente inexistente devido ao *timeout* constantemente recebido caso uma *query* demore muito tempo a ser processada pela base de dados.

Outro exemplo de problema de d2rq que limita de certa forma as funcionalidades espaciotemporais de Trontegra é a falta de suporte de GeoSPARQL. Este padrão de acesso a dados permite mais facilmente separar respostas por filtro espacial ou até temporal, o que seria uma mais-valia para Trontegra.

5.2.3 Integrador de Ontologias

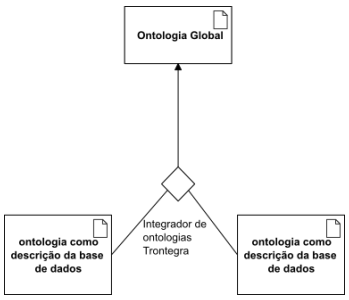

Figura 5.4: Integrador de Ontologias

Com base nas ontologias locais é possível construir uma ontologia global. A sua construção agrega todas as caraterísticas das ontologias locais, permitindo por isso efetuar *queries* gerais de uma forma mais simplificada.

Este módulo foi implementado para funcionar da seguinte forma:

1. Abre cada uma das ontologias locais utilizando Apache Jena. Ao abrir verifica se o servidor correspondente ao ficheiro aberto está a correr, caso contrário irá abortar a integração.

2. Extrai-se a informação acerca da base de dados que se relaciona pela relação **a** à classe **d2rq:Database**, a informação relativa à tabela que se relaciona pela relação **a** à classe **d2rq:ClassMap** e a informação relativa aos campos, que se relacionam pela relação **a** à classe **d2rq:PropertyBridge**, excluindo os campos que também se relacionam a **rdfs:label** pela relação **d2rq:property**

3. Povoa-se uma ontologia geral já pré-feita, representada na Figura 5.5, onde **data_source** é a classe responsável por guardar todas as caraterísticas das fontes de dados, que pode ter uma ou mais **class**es relacionadas a esta pela relação de propriedades de objeto *hasClass*. **class** por sua vez relaciona-se com **data_source** através da relação de propriedades de objeto *belongsToClass* e relaciona-se com **field**s através da relação *hasField*. A relação inversa dessa é *hasClass* que relaciona **field** com **class**.

4. Depois de realizadas todas as operações acima mencionadas, é gerado um ficheiro **final.ttl** que será utilizado posteriormente, e que efetivamente é a ontologia geral.

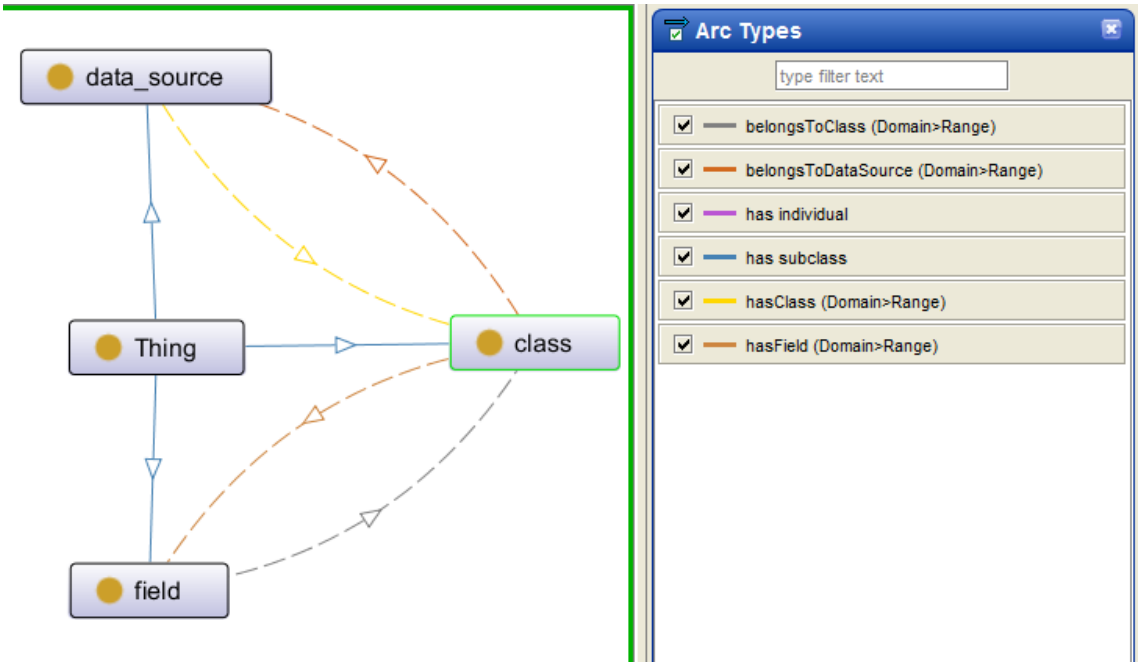

Figura 5.5: Ontologia-base para a construção da ontologia geral

5.2.4 Ponto de Acesso SPARQL Global

Como podemos observar na Figura 5.1, Apache Fuseki possui um Ponto de Acesso SPARQL que em termos gerais permite a um serviço como o Visualizador efetuar *queries* à informação existente nas bases de dados. Para isso é utilizada a informação existente na ontologia geral.

Para colocar em funcionamento este ponto de acesso SPARQL, é inicializado um servidor Apache Fuseki. Como esta ferramenta é um ficheiro binário, similarmente a d2rq é criada uma *thread* que executa esse ficheiro, criando desta forma um novo processo. O número de processo deste também é guardado para posteriormente ser encerrado aquando de Trontegra fechar. Ao inicializar, o processo carrega o ficheiro **final.ttl** e o publica no endereço http://localhost:3030/ds. Juntamente com isto é inicializado o ponto de acesso SPARQL já pronto para ser interrogado pelos serviços.

5.2.5 Serviço de Visualização

O serviço de visualização toma proveito do Ponto de Acesso SPARQL Global para obter a informação que é posteriormente exibida num mapa. Um exemplo de visualização dos dados pode ser visto na Figura 5.6.

Este serviço é implementado usando HTML, Javascript e CSS. Mais especificamente, para a visualização do mapa é utilizada a framework Leaflet. Utilizando tecnologias Web, é possível executar este serviço num *Browser*, tendo esta ferramenta só sido testada no Google Chrome.

Em termos de ordem de funcionamento, esta consiste em interrogar o ponto de acesso SPARQL global, receber a resposta e depois de a analisar e adicionar os dados extraídos desta ao mapa.

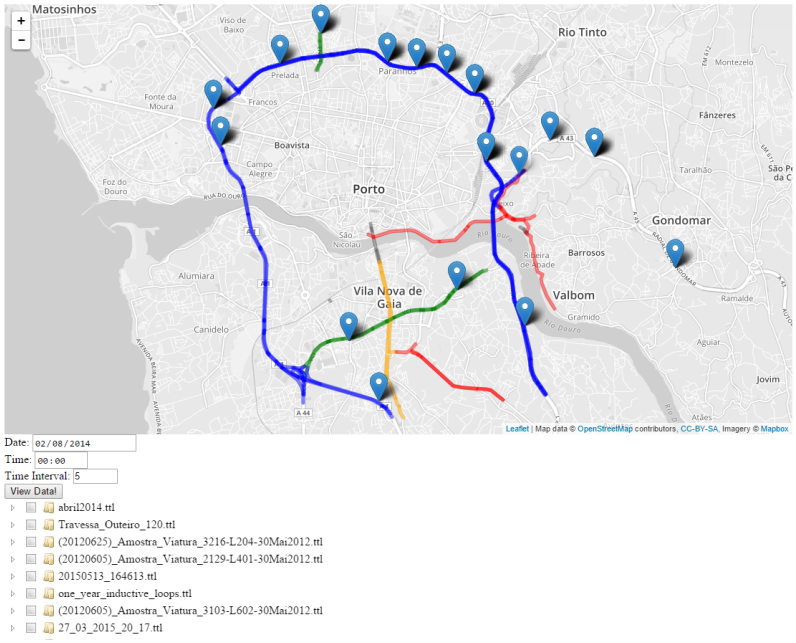

Figura 5.6: Visualizador

5.3 Análise de Resultados

Nesta secção irão ser descritas as caraterísticas do computador onde foram efetuados os testes para analisar a eficiência do Trontegra. Irão também ser explicados em que consistem estes testes e posteriormente é discutida a usabilidade da ferramenta criada e feita uma avaliação acerca dos resultados que se obtiveram com esta.

5.3.1 Descrição da configuração de teste

Os testes são feitos num computador pessoal. As suas caraterísticas são as seguintes:

- Sistema Operativo Windows 8.1 a 64 bits
- Processador Intel i5-3210m a 2.5GHz com 2 núcleos físicos com 2 núcleos virtuais cada
- Memória RAM 8GB DDR3
- Disco de armazenamento SSD Samsung 840 EVO de 250GB

As bases de dados relacionais utilizadas nos testes são as que estão especificadas na Tabela 5.1, incluindo já ficheiros csv e osm importados. No caso destas bases de dados, nomeadamente nas do tipo MySQL, estas possuem maioritariamente dados recolhidas por espiras magnéticas cedidas pela empresa *Armis*. Possuem também alguns *logs* GPS criados a partir de capturas efetuadas com um *smartphone*. No caso de bases de dados PostgreSQL, estas possuem somente dados importador a partir de ficheiros osm, provenientes de OpenStreetMaps.

Tabela 5.1: Bases de Dados Utilizadas para os Testes

Nome da Base de Dados	Nº de Registos Total	Nº de Tabelas	Nº de Colunas Total	Tamanho ocupado em disco
MySQL				
(20120605)_amostra_viatura_2129-l401-30mai2012	1618	1	28	336 KB
(20120605)_amostra_viatura_3103-l602-30mai2012	1612	1	28	336 KB
(20120625)_amostra_viatura_3216-l204-30mai2012	1517	1	28	320 KB
27_03_2015_20_17	416	1	13	64 KB
20150513_164613	3794	1	7	320 KB
abril2014	585817	1	21	76.6 MB
agosto2014	724791	1	21	85.6 MB
one_year_inductive_loops	28750755	1	8	3.4 GB
sensors_location	26	1	6	16 KB
travessa_outeiro_120	723	1	13	96 KB
PostgreSQL				
porto_region	97341	5	282	42MB
vci	36517	5	282	21MB

A execução dos testes foi feita por meio do serviço de visualização construído em JavaScript. Este permite também facilmente selecionar tanto as fontes de dados como os campos de cada fonte de dados que se pretende interrogar, gerando para isso as *queries* correspondentes.

5.3.2 Testes Realizados

Para a verificação da eficiência de funcionamento de Trontegra, executaram-se testes de desempenho. Cada *query* SPARQL foi executada com vários valores para cada variável, como é o caso do número de fontes de dados e o número de campos selecionados. Foram também analisados os resultados provenientes de cada uma destas interrogações. Com base neste conjunto de dados, poderá ser possível tirar conclusões quanto ao funcionamento do sistema.

- **Teste 1** tratou de verificar qual a relação que liga o número de campos selecionados com o tempo que o sistema demora a devolver uma resposta. Este consistiu em criar uma *query* que interrogava uma única fonte de dados, que é a base de dados **abril2014**. Cada passo de incrementação no número dos campos foi testado 10 vezes para despistar possíveis falsos-positivos. O número de resultados devolvidos foi de 90 linhas em todas as interrogações e repetições.

  Depois de se obter o resultado verificou-se através do gráfico da Figura 5.7 que só com um campo selecionado, o tempo-base de resposta é de 1.10 segundos. Apesar de haver

algumas irregularidades no gráfico, e não ser este estritamente linear, pode-se verificar que a diferença entre selecionar um campo ou todos os campos (21 neste caso) de uma fonte de dados faz pouca diferença no tempo de execução. Se se fizer os cálculos, verificar-se-á que esta diferença entre selecionar 1 campo e 21 campos é de 0.72 segundos.

- **Teste 2** aplicou o mesmo conceito do Teste 1, ou seja, incrementar o número de campos selecionados, mas neste caso o número de fontes de dados selecionadas é 8 e não 1, e foram elas todas as bases de dados MySQL tirando a base de dados **one_year_inductive_loops** e a **sensors_location**. A incrementação dos campos selecionados não é feita de forma linear tal como acontecia no teste 1. O número de campos selecionados por cada fonte de dados seguiu esta ordem: 1, 4, 7 e todos os campos. No total, isto representa 4 iterações em que o número de campos selecionados totais foi de 8, 32, 54 e 159. Cada uma das iterações foi repetida 10 vezes para que a probabilidade de haver falsos-positivos seja reduzida. Todas as interrogações obtiveram 1394 linhas de resultados.

  Depois da execução do teste ter terminado, gerou-se o gráfico que está representado na Figura 5.8. Nesta é demonstrado que o tempo de resposta base é de 18.18 segundos para 8 fontes de dados com 8 campos totais selecionadas. O aumento do tempo de resposta à medida que eram feitas as iterações é feito de uma maneira quase linear, sendo que o tempo final com 8 fontes de dados e 159 campos selecionados demora um total de 31.74 segundos a devolver resposta. A diferença de tempo de execução entre a primeira e a última iteração é cerca de 13.5 segundos.

- **Teste 3** já aplica uma forma diferente de alterar as variáveis de teste, que desta vez inclui um número variável de fontes de dados com um número fixo de 4 campos para cada uma destas. Com base nesta abordagem pretendeu-se ver a relação que poderá estar entre o aumento do número de fontes de dados e o tempo de execução de cada *query*. Para complementar isto, também foi adicionada informação acerca do número de resultados que cada interrogação retornou. Igualmente aos outros testes, cada uma das iterações é repetida 10 vezes pelas mesmas razões especificadas acima. As fontes de dados usadas são as mesmas do teste 2.

  Com base nos resultados deste teste, que poderão ser vistos na Figura 5.9, a primeira resposta teve um resultado similar ao do primeiro resultado do teste 1. Em termos de aumento do tempo de execução das *queries* observado, pode-se dizer que este faz lembrar um aumento exponencial. O tempo de interrogação de 1 fonte de dados comparada a 8 fontes de dados é quase 15 vezes menor.

- **Teste 4** por sua vez compara o tempo de resposta de uma fonte de dados com o tamanho em termos do seu número de entradas. As fontes de dados usadas neste teste foram interrogadas por ordem crescente de grandeza, uma a uma. Cada uma delas tinha somente um campo selecionado.

  Os resultados deste teste estão na Figura 5.10. É de notar que se formos comparar os tempos de execução, não há variações maiores do que 1.5 segundos entre estes.

Embora seja um termo de comparação válido, não foi possível controlar a *query* de maneira a que o conjunto de resultados fosse parecido entre as fontes de dados, daí que nem todas elas estejam incluídas neste teste, como é o caso de **one_year_inductive_loops**. No caso desta última em vez de ser obtido um resultado, era obtido um erro por causa do seu tamanho.

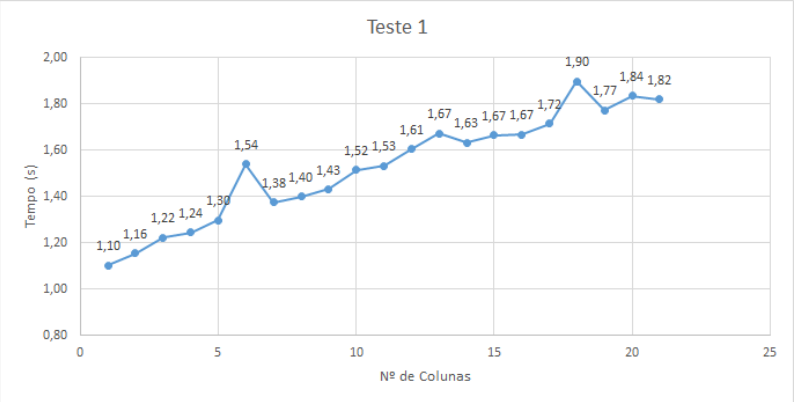

Figura 5.7: Medição de Tempo de Resposta com nº variado de campos selecionados - 1 fonte de dados

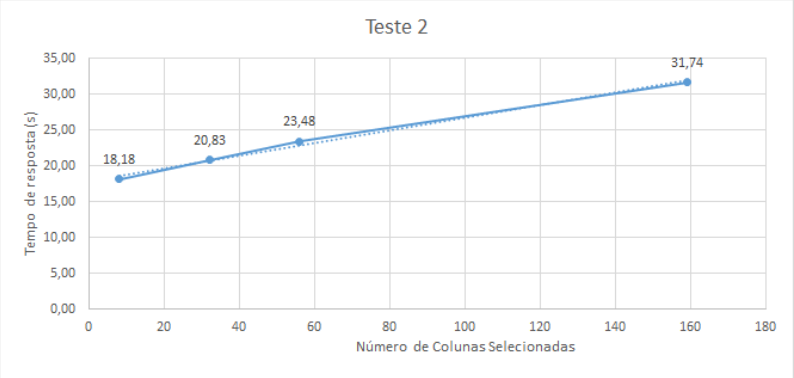

Figura 5.8: Medição de Tempo de Resposta com nº variado de campos selecionadas - 8 fontes de dados

5.3.3 Discussão

Depois de analisar os resultados provenientes dos testes efetuados ao sistema, existem vários pontos que precisam de ser anotados e explicados. Um destes é o propósito para o qual se usa Trontegra:

- Caso seja necessário um sistema que suporte dados gerados em tempo real, Trontegra não irá satisfazer estas necessidades. Isto acontece porque ao adicionar um modelo de dados intermédio entre os dados reais e o utilizador, irá sempre haver atraso tanto no processamento de resultados como em tradução de SPARQL para SQL.

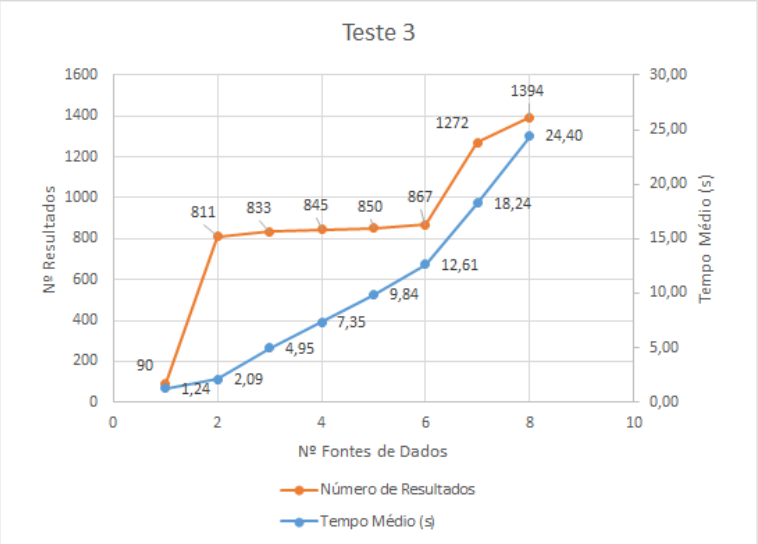

Figura 5.9: Medição de Tempo de Resposta e do Número de Resultados com nº variado de fontes de dados

Existem formas de evitar a existência deste *overhead*. Uma delas é a conversão de todos os dados existentes nas fontes de dados para o formato RDF. Isto terá de ser feito antes de adicionar a fonte de dados ao sistema. Desta forma as *queries* SPARQL serão diretamente executadas na sua forma original sobre os dados sem a necessidade de existir um processo de tradução. A desvantagem deste método reside no grande tempo de processamento inicial necessário para a conversão de dados entre formatos e no vasto espaço de armazenamento necessário para alojar os dados convertidos. Por essa razão a forma seguinte poderá ser mais vantajosa. Esta consiste em usar fontes de dados que armazenem os seus dados diretamente no formato RDF. Desta forma, por lado do Trontegra não haverá *overhead* nem na conversão dos dados nem na tradução das *queries* de SPARQL para SQL e não será necessário alojar os dados provenientes das fontes de dados localmente onde Trontegra se encontra em execução.

- Caso o serviço necessite de ter a possibilidade de integrar várias fontes de dados num único ponto de acesso, e não precisar de grande eficiência no retorno de respostas, Trontegra já servirá para este propósito.

Tendo mencionado isto, pode-se delinear as qualidades e limitações do Trontegra, tendo por base os testes realizados e a utilização deste no dia-a-dia. Como suas qualidades tem-se:

- A possibilidade de adicionar novas fontes de dados ao sistema de uma forma simples e transparente ao utilizador.
- A facilidade de interrogar várias fontes de dados simultaneamente.
- O acesso aos dados presentes no sistema é feito de forma simples através de uma consola SPARQL, o que facilita a construção de serviços que queiram utilizar estes dados.

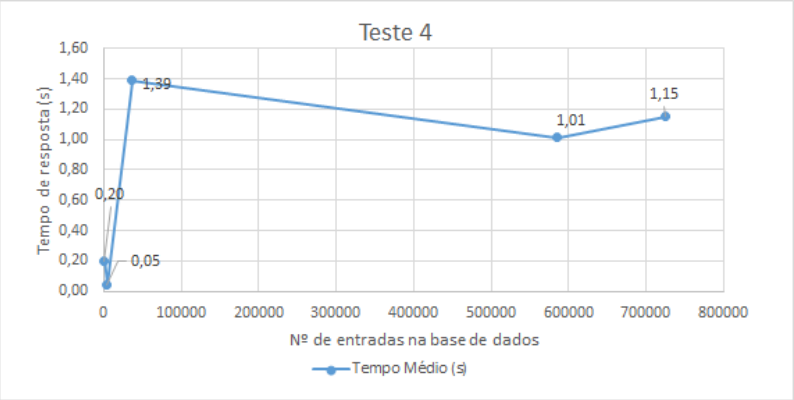

Figura 5.10: Medição de Tempo de Resposta com base no nº de entradas existentes na fonte de dados

- Em termos de quantidade de campos selecionados por cada fonte de dados, o seu número não aumenta o tempo de execução da *query* de uma forma notória. Tal como vimos no teste 1, a diferença de interrogar uma fonte de dados com 1 campo ou 21 campos é de 0.72 segundos no tempo total.

Em termos de limitações existentes:

- Por agora Trontegra só está disponível para o sistema operativo Windows.
- Tem um suporte limitado a formatos de novas fontes de dados.
- Sendo baseado em ferramentas *open-source*, algumas destas estão desatualizadas que fazem com que Trontegra fique limitado em certas funcionalidades. Exemplo disso é a impossibilidade de interrogar bases de dados muito grandes, que é um problema especificamente da ferramenta d2rq.
- Em seguimento do ponto anterior, e apesar de não se ter visto um grande aumento de tempo de resposta entre interrogar uma fonte de dados com 1600 entradas e 500000, a interrogação da base de dados que tem 27 milhões de entradas (one_year_inductive_loops) não é possível pois o Trontegra ao fim de 60 segundos simplesmente recebia uma mensagem de tempo limite esgotado, gerado por d2rq.
- Impossibilidade de suporte ao acesso simultâneo à mesma fonte de dados por vários utilizadores. O que acontece atualmente é se um utilizador está à espera de receber uma resposta a partir de uma fonte de dados e ao mesmo tempo caso um outro utilizador queira interrogar essa mesma fonte de dados, este não o vai poder efetuar até o primeiro receber a resposta.
- A execução desta ferramenta com muitas fontes de dados presentes implica uma grande utilização de memória RAM do computador em que Trontegra está a ser executado. Isto porque para cada uma das fontes de dados é necessário correr um servidor d2rq, que em média ocupa 100MB de RAM.

Por causa de Trontegra ser em termos gerais uma camada intermédia entre o utilizador e os dados, a sua eficiência nunca será melhor do que duma base de dados relacional. Isto acontece porque ao se usar este sistema, existem conversões de dados a decorrer no processo de uma interrogação, que aumentam o tempo de espera por uma resposta. Daí se pode concluir que sendo Trontegra um facilitador de acesso a dados provenientes de fontes diferentes, isso não faz deste sistema mais eficiente que uma solução desenvolvida de raiz para servir um determinado modelo de dados já pré-conhecido aquando da construção deste.

5.4 Sumário

Neste capítulo foram abordados muitos tópicos importantes. Em termos de organização da plataforma, foi explicada a sua arquitetura, que permite entender como está estruturada a ferramenta Trontegra, quais são os seus módulos principais e como estes comunicam uns com os outros. Foi também explicada de que maneira Trontegra foi implementada, mencionando também a forma usada de conceção e construção. São igualmente indicadas as principais tecnologias e ferramentas utilizadas na construção desta ferramenta.

Foram demonstrados os módulos do sistema, sendo feita uma análise separada para cada um destes para uma visão e perceção mais fácil do sistema. Na parte final foram demonstrados os testes realizados que serviram para verificar a eficiência e a facilidade de utilização da ferramenta criada. São também abordados os casos nos quais seria mais ou menos útil utilizar este sistema. Em termos de discussão dos resultados obtidos com base nos testes realizados à ferramenta, foi concluído que Trontegra é para ser utilizado por sistemas que não precisem de dados em tempo real. Em termos de tamanho das bases de dados utilizadas, recomenda-se que estas não ultrapassem o milhão de registos, de forma a que a resposta às *queries* não seja demasiado demorada.

Capítulo 6

Conclusões e Trabalho Futuro

Neste capítulo é feito o resumo de todo o trabalho apresentado nesta dissertação. Também são apresentadas as contribuições científicas principais que o trabalho trouxe e são analisados quais os objetivos que foram ou não alcançados. Por fim são discutidos possíveis trabalhos futuros que possam de alguma forma ter origem no trabalho desenvolvido nesta dissertação.

6.1 Resumo geral

Neste documento foi inicialmente introduzido o contexto do trabalho, qual o motivo e os objetivos definidos para este. Em termos de contexto, é um trabalho desenvolvido na área de análise e processamentos de dados espaciotemporais que tem como motivação a grande diversidade de dados e fontes de dados que dificultam a sua análise. Sendo portanto o objetivo encontrar uma forma de o efetuar a partir de uma só ferramenta. Esta tem que ser capaz de integrar as variadas fontes de dados e as partilhar num ponto de acesso para que serviços possam aí aceder e consumir os dados.

A seguir a isto foi feita a revisão de conceitos. Este capítulo ajuda a quem esteja a ler este documento a entender melhor o vocabulário que aqui é utilizado. Para a construção deste foi feita uma revisão de literatura relacionada com este trabalho, que foi abordada mais a fundo no capítulo seguinte. Esta análise foi construída introduzindo inicialmente os conceitos mais genéricos e posteriormente estreitou o seu campo de visão para conceitos mais diretamente incidentes e que têm uma importância maior no trabalho elaborado. Estes últimos são relacionados com ontologia e a arquitetura orientada a serviços.

No capítulo 3 foi dado um grande foco nos trabalhos relacionados, tanto na forma de artigos já publicados como de ferramentas já construídas. No primeiro caso, os artigos mais importantes foram o **Ontology-Based Integration of Data Sources** [Gag07] do qual se tirou a ideia das abordagens sobre as fontes de dados e foi sobre a arquitetura do sistema proposto neste artigo que

é baseada a construção de Trontegra. Outros artigos importantes são **Ontology-based geospatial data query and integration**[ZZWP08] e **Quality Aware Service Oriented Ontology Based Data Integration** [HC13]. Em termos de ferramentas relevantes encontrou-se o Mastro e a ferramenta construída pelo autor do artigo [ZZWP08]. Ambas aplicam o modelo ontológico sobre dados, mas só a segunda o aplica a dados espaciotemporais. Mas isto é feito com algumas lacunas, as quais o Trontegra tenta corrigir.

No capítulo seguinte é formalmente apresentado o problema que Trontegra tenta resolver. Ao invés do primeiro capítulo, este explica mais aprofundadamente que a grande dispersão de fontes de dados dificulta a agregação e posterior análise destas. Depois disto é apresentada uma solução genérica e ideal, que teoricamente resolverá o problema apresentado anteriormente. É uma solução com 4 camadas principais, tendo cada uma delas um objetivo bem especificado e definido. São elas A - conjunto de fontes de dados, B - descritor semântico, C - integrador e D - distribuidor dos dados. Resumindo muito brevemente cada uma das camadas, poder-se-á dizer que a camada A é responsável por gerir as fontes de dados existentes no sistema, B pega em cada uma destas e as descreve para forma de uma ontologia, para ser utilizada pelas camadas seguintes. Na camada C é efetuada a junção de todas as descrições ontológicas das fontes de dados, gerando como produto final a ontologia geral. Na camada D, última do ponto de vista do sistema e primeira da perspetiva de um utilizador, é usada a ontologia geral e são distribuídos os dados entre os serviços que os queiram utilizar. A seguir a explicar a solução genérica são apresentados possíveis casos de utilização de tal sistema.

Tendo isto em mente, é de seguida apresentada a abordagem escolhida para elaborar a prova de conceito da solução genérica proposta, que é o sistema Trontegra. Esta segue a mesma estrutura das 4 camadas, mas define limites para as possibilidades que foram apresentadas na solução genérica. Exemplos disso são os formatos de fontes de dados aceites, a semântica que descreve cada campo, cada fonte de dados ser bastante limitada e o acesso de utilizadores aos dados simultaneamente não ser possível.

No capítulo 5 é finalmente abordada a implementação do sistema Trontegra e posteriormente são-lhe feitos testes de eficiência, discutindo de seguida os resultados agregados. Na implementação é discutida a arquitetura do sistema, de como esta tem 4 módulos e como cada um foi implementado. Adicionalmente também é demonstrado o funcionamento de um serviço que implementa a possibilidade de visualizar graficamente os dados presentes no sistema. É também dado foco nas tecnologias usadas e posteriormente são abordadas as ferramentas utilizadas, tais como d2rq, Apache Jena e Leaflet. De seguida são apresentados os testes realizados a Trontegra e feita uma discussão sobre os resultados obtidos. Em termos gerais, Trontegra não serve para serviços que queiram dados em tempo real nem que pretendam fazer *queries* a bases de dados relacionais com muitas entradas, mas serve sim para serviços que pretendam ter acesso a dados de variadas fontes através de um só ponto de acesso sem a necessidade de os ter o quanto antes.

Neste capítulo é resumido todo o trabalho descrito nesta dissertação, é discutida a satisfação dos objetivos e dada uma perspetiva futurista acerca de trabalhos que eventualmente possam surgir a partir deste.

6.2 Discussão das perguntas de investigação

Depois de se ter efetuado a pesquisa na literatura e posteriormente se ter construído a ferramenta pretendida, já se possui conhecimento suficiente para responder às perguntas colocadas inicialmente.

1. Como interoperabilizar múltiplas perspetivas de análise de transporte/tráfego a partir de *Open Data*?

   Para responder melhor a esta questão será necessário efetuar testes de usabilidade ao serviço de visualização. No início houve planos para desenvolver este serviço mas posteriormente verificou-se que esta pergunta fugia ao âmbito deste trabalho. Deste modo esta é uma questão que ficará para um eventual trabalho futuro.

2. Qual o método de junção de fontes de dados a utilizar para uma eficiente obtenção e processamento de dados rodoviários (*data integration*; *sensor integration*)?

   Com base na elaboração deste projeto, verificou-se realmente que o método ontológico facilita a integração de variadas fontes de dados, mas não torna o processo de obtenção de dados eficiente.

3. Quais os métodos de junção de fontes de dados, para a questão acima, tendo por base um modelo ontológico?

   Os métodos encontrados baseiam-se nas ideias de Michel Gagnon [Gag07], que apresenta várias arquiteturas de integração de fontes de dados através de ontologia. Estes encontram-se resumidos na Secção 3.2.1.

4. Como automatizar o processo de adição de uma nova fonte de informação ao já existente modelo ontológico?

   Para que este processo fosse automático, foi necessário construir uma ferramenta dedicada a isso. Esta atualmente é capaz de receber parâmetros básicos do utilizador e com base nisso efetuar os passos necessários para que fosse feito o mapeamento de uma fonte de dados para uma ontologia local. De seguida esta mesma ferramenta atualiza o modelo ontológico geral com a informação presente na ontologia local criada.

5. De que forma será possível aplicar uma arquitetura orientada a serviços ao modelo ontológico que se pretende aplicar, de modo a permitir outras entidades acederem aos dados?

   Para responder a esta questão, foi analisada a literatura relacionada mas nada foi encontrado que relacionasse o conceito de ontologia com a arquitetura orientada a serviços. Por essa razão a arquitetura da solução proposta que foi desenvolvida neste projeto é uma contribuição científica, que possui como contrato a ontologia geral. Esta permite a serviços saberem como estão organizadas tanto as fontes de dados bem como os seus dados. Estes com base nessa informação poderão formular as suas próprias interrogações e obter dados.

6. Quão eficientes são as interrogações num sistema baseado em integração por ontologia?

   Tal como foi analisado nos 4 testes feitos ao funcionamento do sistema (Secção 5.3.2), a sua eficiência é pobre. Esta caraterística é mais notável quando são interrogadas várias fontes de dados simultaneamente.

6.3 Satisfação dos Objetivos

Nesta secção é agora feito um balanço acerca das contribuições deste trabalho. Mais especificamente irá ser analisada a satisfação dos seus objetivos.

Em termos de metas concluídas, foi realmente construído um sistema que permite adicionar novas fontes de dados. Tal como mencionado no Capítulo 5, os formatos das fontes de dados suportados são csv, osm.pbf e bases de dados relacionais.

Outro ponto que foi alcançado é a criação de uma ontologia individual/local que descreva detalhadamente cada uma das fontes de dados. Em termos deste objetivo há porções deste que não foram alcançados, que é o caso da descrição semântica de cada uma das fontes de dados utilizando ontologias externas ou até redes semânticas. Eventualmente estas poderiam permitir ao utilizador escolher para que propósito cada uma das fontes de dados serve e o que cada um dos campos desta significam exatamente. Este é um objetivo parcial que poderá ser aplicado num trabalho futuro.

Ainda em termos da ontologia local, a sua alocação em tempo de execução é localizada num servidor d2rq. Apesar de esta ser uma boa ferramenta que permite fazer tanto mapeamentos como conversões de SPARQL para SQL e vice-versa, o seu suporte para a versão SPARQL mais recente está limitada pelo uso de Joseki no seu interior. Com isto ainda presente, o seu suporte a fontes de dados com número de entradas muito grandes é inexistente.

Em termos de integração das variadas ontologias locais, esta realmente acontece, e é gerada uma ontologia geral que posteriormente é usada para interrogar várias fontes de dados simultaneamente. A limitação deste processo centra-se no facto de ao nível da integração de ontologias não ser feita nenhuma correspondência de campos nem de fontes de dados. Atualmente este processo de correspondência é feito ao nível das *queries* criadas pelo utilizador, sendo por isso considerado um processo manual e não automático. O método automático poderá ser assunto a tratar num trabalho futuro, em que existe implementada uma rede semântica no sistema para uma mais fácil deteção de campos ou fontes parecidas. Caso não haja, é possível ser feita a correspondência com base na comparação da designação textual (em inglês *string comparison*), usando para isso algoritmos que medem a distância entre duas *strings*.

A distribuição dos dados ocorre por meio de uma consola SPARQL. Para que se possa obter dados é necessário consultar a ontologia geral que permite ver a organização destes. Tendo estes fatores em conta, pode-se considerar que se tem um conceito não completo mas parecido com a arquitetura orientada a serviços, em que a ontologia geral é o contrato que os serviços têm que obter, ler, e conforme este interrogar os dados. Por isso a parte de ter uma arquitetura orientada a serviços é um objetivo cumprido.

Um último objetivo que foi também satisfeito é a construção de um serviço de visualização dos dados que usa o sistema Trontegra. Possui a possibilidade de ver um mapa povoado com dados acerca da topologia das estradas da VCI e também a localização e medições das espiras magnéticas.

6.4 Trabalho Futuro

Depois de terminada a construção do sistema proposto, existe espaço para melhorias e extensões de limitações que ainda persistem. As possibilidades são:

- Trocar a ferramenta d2rq por alguma que seja mais atualizada, que seja mais eficiente tanto em termos de tradução de SPARQL para SQL e vice-versa. Se possível também com suporte a GeoSPARQL.
- Ao sistema já existente expandir o público-alvo em termos de formatos de fontes de dados.
- Adicionar possibilidade de para além de ter um ponto de acesso SPARQL, poder extrair dados existentes no sistema como um ficheiro para o formato aceite por um simulador do tipo SUMO.
- Adicionar à ferramenta existente uma rede semântica que se poderá ligar ao modelo extraído de cada fonte de dados. Desta forma será possível descrever semanticamente uma fonte de dados e os seus campos. Neste caso já se poderá utilizar o formato OWL que permitirá ter uma maior expressividade em termos semânticos. Com isto espera-se que agentes automáticos se possam ligar ao sistema e conseguirem saber o significado dos dados.
- Seria útil adicionar ao sistema na parte da integração das várias ontologias o encontro de correspondências e conexões entre campos e fontes de dados que de alguma forma possam ser semelhantes. Este processo tanto pode ser baseado no ponto anterior como em algoritmos que calculam a distância entre duas *strings*.
- A ideia atual de aplicar um modelo ontológico a dados espaciotemporais pode ser estendida para outros conjuntos de dados, como medicina, linguística, etc.
- Alocar o sistema na *Cloud* de forma a ter uma maior escalabilidade de recursos utilizados por este.

Finalmente, este trabalho irá integrar a arquitetura MAS-Ter Lab, em desenvolvimento no LIACC, constituindo uma plataforma para a análise e gestão integradas de sistemas de transportes [ROB07, RFBO08]. Esta plataforma é concebida tendo como base o conceito de sistemas artificiais de transportes (*Artificial Transportation Systems*) [RLT11, RL14] e oferecerá, entre outros serviços, ferramentas de controlo e gestão de tráfego [RFBO08], integração e interoperação de simuladores com diferentes resoluções [PRK11, MKS+13], assim como a monitorização de transportes multimodais [ZRC14].

Bibliografia

[ABM08] Serge Abiteboul, Omar Benjelloun e Tova Milo. The active xml project: an overview. *The VLDB Journal*, 17(5):1019–1040, 2008.

[AG00] Jose Maria Abasolo e Mario Gomez. Melisa: An ontology-based agent for information retrieval in medicine. In *Proceedings of the first international workshop on the semantic web (SemWeb2000)*, pages 73–82, 2000.

[AGR+13] Patricia RJ Alves, Joaquim Goncalves, Rosaldo JF Rossetti, Eugenio C Oliveira, Cristina Olaverri-Monreal et al. Forward collision warning systems using heads-up displays: Testing usability of two new metaphors. In *Intelligent Vehicles Symposium Workshops (IV Workshops), 2013 IEEE*, pages 1–6. IEEE, 2013.

[AJ11] Benjamin Adams e Krzysztof Janowicz. Constructing geo-ontologies by reification of observation data. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 309–318. ACM, 2011.

[AO05] Hilary Arksey e Lisa O'Malley. Scoping studies: towards a methodological framework. *International journal of social research methodology*, 8(1):19–32, 2005.

[ARFC15] João E Almeida, Rosaldo JF Rossetti, Brígida Mónica Faria e António Leça Coelho. Using serious games to train children and elicit fire safety behaviour. In *New Contributions in Information Systems and Technologies*, pages 1153–1162. Springer, 2015.

[AWS20] K. Ahmad, T.S.F.T. Wook e R. Samad. Key based approach for integration of heterogeneous data sources. *Journal of Theoretical and Applied Information Technology*, 48(2):699 – 703, 2013/02/20. heterogeneous data sources integration;key based approach;patients medical records;medical data storage;predefined global schema;analysis phase;design phase;implementation phase;testing phase;database information;.

[CDGL+11] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, Riccardo Rosati, Marco Ruzzi e Domenico Fabio Savo. The mastro system for ontology-based data access. *Semantic Web*, 2(1):43–53, 2011.

[CGY07] Nadine Cullot, Raji Ghawi e Kokou Yétongnon. Db2owl: A tool for automatic database-to-ontology mapping. In *SEBD*, pages 491–494, 2007.

[Col14] Pieter Colpaert. Route planning using linked open data. In *The Semantic Web: Trends and Challenges*, pages 827–833. Springer, 2014.

[CRF03] William Cohen, Pradeep Ravikumar e Stephen Fienberg. A comparison of string metrics for matching names and records. In *Kdd workshop on data cleaning and object consolidation*, volume 3, pages 73–78, 2003.

[CS14] Michał Chromiak e Krzysztof Stencel. A data model for heterogeneous data integration architecture. In *Beyond Databases, Architectures, and Structures*, pages 547–556. Springer, 2014.

[CSR10] Sara Filipa Lemos de Carvalho, Luis Sarmento e Rosaldo J.F. Rossetti. Real-time sensing of traffic information in twitter messages. In *Proceedings of the IEEE ITSC 2010 Workshop on Artificial Transportation Systems and Simulation (ATSS'2010), Madeira Island, Portugal*, 2010.

[CTL12] Shih-Wei Chen, Yu-Ting Tseng e Tsai-Ya Lai. The design of an ontology-based service-oriented architecture framework for traditional chinese medicine healthcare. In *e-Health Networking, Applications and Services (Healthcom), 2012 IEEE 14th International Conference on*, pages 353–356. IEEE, 2012.

[CW11] Noel Cressie e Christopher K Wikle. *Statistics for spatio-temporal data*. John Wiley & Sons, 2011.

[Due79] Kenneth J Dueker. Land resource information systems: a review of fifteen years experience. *Geo-Processing (Netherlands)*, 1979.

[EFLK11] Nour-Eddin El Faouzi, Henry Leung e Ajeesh Kurian. Data fusion in intelligent transportation systems: Progress and challenges–a survey. *Information Fusion*, 12(1):4–10, 2011.

[FCR09] Tiago RM Freitas, António Coelho e Rosaldo JF Rossetti. Improving digital maps through gps data processing. In *Intelligent Transportation Systems, 2009. ITSC'09. 12th International IEEE Conference on*, pages 480–485. IEEE, 2009.

[FCR10] Tiago RM Freitas, António Coelho e Rosaldo JF Rossetti. Correcting routing information through gps data processing. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 706–711. IEEE, 2010.

[FE99] Frederico T Fonseca e Max J Egenhofer. Ontology-driven geographic information systems. In *Proceedings of the 7th ACM international symposium on Advances in geographic information systems*, pages 14–19. ACM, 1999.

[FRK+14] João Filgueiras, Rosaldo JF Rossetti, Zafeiris Kokkinogenis, Michel Ferreira, Cristina Olaverri-Monreal, Marco Paiva, João Manuel RS Tavares e Joaquim Gabriel. Sensing bluetooth mobility data: potentials and applications. In *Computer-based Modelling and Optimization in Transportation*, pages 419–431. Springer, 2014.

[Gag07] Michel Gagnon. Ontology-based integration of data sources. In *Information Fusion, 2007 10th International Conference on*, pages 1–8. IEEE, 2007.

[GBMP13] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic e Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660, 2013.

[GGROM14] Joaquim Goncalves, Joao SV Goncalves, Rosaldo JF Rossetti e Cristina Olaverri-Monreal. Smartphone sensor platform to study traffic conditions and assess driving performance. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 2596–2601. IEEE, 2014.

[GJR+15] João SV Gonçalves, João Jacob, Rosaldo JF Rossetti, António Coelho e Rui Rodrigues. An integrated framework for mobile-based adas simulation. In *Modeling Mobility with Open Data*, pages 171–186. Springer, 2015.

[GRJ+14] Joao SV Goncalves, Rosaldo JF Rossetti, Jeevamma Jacob, Joaquim Goncalves, Cristina Olaverri-Monreal, Antonio Coelho e Rodrigo Rodrigues. Testing advanced driver assistance systems with a serious-game-based human factors analysis suite. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*, pages 13–18. IEEE, 2014.

[GROM12] Joel Gonçalves, Rosaldo JF Rossetti e Cristina Olaverri-Monreal. Ic-deep: A serious games based application to assess the ergonomics of in-vehicle information systems. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 1809–1814. IEEE, 2012.

[Gru93] Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.

[Gru95] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5):907–928, 1995.

[HC13] MS Hema e S Chandramathi. Quality aware service oriented ontology based data integration. *WSEAS Transactions on Computers*, 12(12):463–473, 2013.

[HWSW11] Siquan Hu, Haiou Wang, Chundong She e Junfeng Wang. Agont: Ontology for agriculture internet of things. In *Computer and Computing Technologies in Agriculture IV*, pages 131–137. Springer, 2011.

[JfWmWd+06] Song Jun-feng, Zhang Wei-ming, Xiao Wei-dong, Tang Da-quan e Tang Jiu-yang. Research on constructing ontology for the semantic web. In *Emerging Trends and Challenges in Information Technology Management: 2006 Information Resources Management Association International Conference, Washington, DC, USA, May 21-24, 2006*, volume 1, page 55. IGI Global, 2006.

[KFC+15] Zafeiris Kokkinogenis, João Filguieras, Sara Carvalho, Luís Sarmento e Rosaldo J.F. Rossetti. Chapter 12 - mobility network evaluation in the user perspective: Real-time sensing of traffic information in twitter messages. In Rosaldo J.F. RossettiRonghui Liu, editor, *Advances in Artificial Transportation Systems and Simulation*, pages 219 – 234. Academic Press, Boston, 2015. URL: http://www.sciencedirect.com/science/article/pii/B9780123970411000121, doi:http://dx.doi.org/10.1016/B978-0-12-397041-1.00012-1.

[KMG06] Lawrence A Klein, Milton K Mills e David RP Gibson. Traffic detector handbook: -volume ii. Technical report, U.S. Department of Transportation, 2006.

[KVS+] Kostis Kyzirakos, Ioannis Vlachopoulos, Dimitrianos Savva, Stefan Manegold e Manolis Koubarakis. Geotriples: a tool for publishing geospatial data as rdf graphs using r2rml mappings.

[LRB09] Pedro FQ Loureiro, Rosaldo JF Rossetti e Rodrigo AM Braga. Video processing techniques for traffic information acquisition using uncontrolled video streams. In *Intelligent Transportation Systems, 2009. ITSC'09. 12th International IEEE Conference on*, pages 127–133. IEEE, 2009.

[MGR91] David J Maguire, Michael F Goodchild e David W Rhind. *Principles and Applications*. Longman, 1991.

[MKS+13] Joao Macedo, Zafeiris Kokkinogenis, Gustavo Soares, Deborah Perrotta e Rosaldo JF Rossetti. A hla-based multi-resolution approach to simulating electric vehicles in simulink and sumo. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 2367–2372. IEEE, 2013.

[MPS+14] Silvia Mirri, Catia Prandi, Paola Salomoni, Franco Callegati e Aldo Campi. On combining crowdsourcing, sensing and open data for an accessible smart city. In *Next Generation Mobile Apps, Services and Technologies (NGMAST), 2014 Eighth International Conference on*, pages 294–299. IEEE, 2014.

[NB01] Tom Nishida e Anthony J Booth. Recent approaches using gis in the spatial analysis of fish populations. *Spatial processes and management of marine populations. Alaska Sea Grant College Program AK-86-01-02*, pages 19–36, 2001.

[NM+01] Natalya F Noy, Deborah L McGuinness et al. Ontology development 101: A guide to creating your first ontology, 2001.

[OMR14] Cristina Olaverri Monreal e Rosaldo JF Rossetti. Human factors in intelligent vehicles [guest editorial]. *Intelligent Transportation Systems, IEEE Transactions on*, 15(4):1734–1737, 2014.

[Ope] Open Knowledge Foundation. What is Open Data? - Open Data Handbook. Acedido: Fevereiro 2015. URL: http://opendatahandbook.org/en/what-is-open-data/.

[Pan14] Abhishek Pandey. *Relational Schema Integration Using Ontologies*. PhD thesis, University of Cincinnati, 2014.

[PLT08] Weidong Pan, Jixue Liu e Jiashen Tian. An implementation of xml data integration. volume DISI, pages 111 – 116, Barcelona, Spain, 2008. Data integration;Data source;Data transformation;Enterprise information integration;Enterprise information system;Global schemas;Required functionalities;Research groups;XML data;.

[PRK11] Lúcio Sanchez Passos, Rosaldo JF Rossetti e Zafeiris Kokkinogenis. Towards the next-generation traffic simulation tools: a first appraisal. In *Information Systems and Technologies (CISTI), 2011 6th Iberian Conference on*, pages 1–6. IEEE, 2011.

[PvdH03] Mike P Papazoglou e Willem-Jan van den Heuvel. Service-oriented computing: State-of-the-art and open research issues. *IEEE Computer. v40 i11*, 2003.

[RAKG13] Rosaldo JF Rossetti, Joao Emilio Almeida, Zafeiris Kokkinogenis e Joel Gonçalves. Playing transportation seriously: Applications of serious games to artificial transportation systems. *IEEE Intelligent Systems*, (4):107–112, 2013.

[RAR+12] João Ribeiro, João Emílio Almeida, Rosaldo JF Rossetti, António Coelho e António Leça Coelho. Using serious games to train evacuation behaviour. In *Information Systems and Technologies (CISTI), 2012 7th Iberian Conference on*, pages 1–6. IEEE, 2012.

[RB99] Rosaldo José Fernandes Rossetti e Sergio Bampi. A software environment to integrate urban traffic simulation tasks. *Journal of Geographic Information and Decision Analysis*, 3(1):56–63, 1999.

[RFBO08] Rosaldo JF Rossetti, Paulo AF Ferreira, Rodrigo AM Braga e Eugénio C Oliveira. Towards an artificial traffic control system. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 14–19. IEEE, 2008.

[RL05] Rosaldo JF Rossetti e Ronghui Liu. An agent-based approach to assess drivers' interaction with pre-trip information systems. In *Intelligent Transportation Systems*, volume 9, pages 1–10. Taylor & Francis, 2005.

[RL14] Rosaldo JF Rossetti e Ronghui Liu. *Advances in Artificial Transportation Systems and Simulation*. Academic Press, 2014.

[RLT11] Rosaldo JF Rossetti, Ronghui Liu e Shuming Tang. Guest editorial special issue on artificial transportation systems and simulation. *IEEE Transactions on Intelligent Transportation Systems*, 2(12):309–312, 2011.

[ROB07] Rosaldo JF Rossetti, Eugénio C Oliveira e Ana LC Bazzan. Towards a specification of a framework for sustainable transportation analysis. In *13th Portuguese Conference on Artificial Intelligence, Guimarães, Portugal*. Citeseer, 2007.

[RPKC11] Catherine Roussey, Francois Pinet, Myoung Ah Kang e Oscar Corcho. An introduction to ontologies and ontology engineering. In *Ontologies in Urban Development Projects*, pages 9–38. Springer, 2011.

[SAT+12] François Scharffe, Ghislain Atemezing, Raphaël Troncy, Fabien Gandon, Serena Villata, Bénédicte Bucher, Fayçal Hamdi, Laurent Bihanic, Gabriel Képéklian, Franck Cotton et al. Enabling linked data publication with the datalift platform. In *Proc. AAAI workshop on semantic cities*, pages No–pagination, 2012.

[Sav09] Alexandr Savinov. Concept-oriented model. *Encyclopedia of Database Technologies and Applications*, 2009.

[SBD13] Rashed Salem, Omar Boussaïd e Jérôme Darmont. Active xml-based web data integration. *Information Systems Frontiers*, 15(3):371–398, 2013.

[SE90] Jeffrey Star e John Estes. Geographic information systems. *An Introduction. Englewood Cliffs, New Jersey (USA)*, 1990.

[SMSE87] Terence R Smith, Sudhakar Menon, Jeffrey L Star e John E Estes. Requirements and principles for the implementation and construction of large-scale geographic

information systems. *International Journal of Geographical Information System*, 1(1):13–31, 1987.

[SNF12] Heiner Stuckenschmidt, Jan Noessner e Faraz Fallahi. A study in user-centric data integration. In *ICEIS (3)*, pages 5–14, 2012.

[STBW02] Ralf-Peter Schäfer, Kai-Uwe Thiessenhusen, Elmar Brockfeld e Peter Wagner. A traffic information system by means of real-time floating-car data. In *ITS World Congress 2002*, 2002.

[TRC+] Joe Tekli, Antoine Abou Rjeily, Richard Chbeir, Gilbert Tekli, Pélagie Houngue, Kokou Yetongnon e Minale Ashagrie Abebe. Semantic to intelligent web era.

[VBGK09] Julius Volz, Christian Bizer, Martin Gaedke e Georgi Kobilarov. Silk-a link discovery framework for the web of data. *LDOW*, 538, 2009.

[VOL+11] Stijn Verstichel, Femke Ongenae, Leanneke Loeve, Frederik Vermeulen, Pieter Dings, Bart Dhoedt, Tom Dhaene e Filip De Turck. Efficient data integration in the railway domain through an ontology-based methodology. *Transportation Research Part C: Emerging Technologies*, 19(4):617–643, 2011.

[VSWV00] Ubbo Visser, Heiner Stuckenschmidt, Holger Wache e Thomas Vögele. Enabling technologies for interoperability. In *Workshop on the 14th International Symposium of Computer Science for Environmental Protection*, pages 35–46. Citeseer, 2000.

[W3C12] W3C. OWL 2 Web Ontology Language New Features and Rationale (Second Edition), 2012. Acedido: 14 Fevereiro 2015. URL: http://www.w3.org/TR/2012/REC-owl2-new-features-20121211/.

[Wan97] Huaiqing Wang. A conceptual model for virtual markets. *Information & Management*, 32(3):147–161, 1997.

[Wor] World Wide Web Consortium. W3C Semantic Web FAQ. Acedido: Fevereiro 2015. URL: http://www.w3.org/2001/sw/SW-FAQ.

[WSSKR99] Holger Wache, Th Scholz, Helge Stieghahn e Brigitta Konig-Ries. An integration method for the specification of rule-oriented mediators. In *Database Applications in Non-Traditional Environments, 1999.(DANTE'99) Proceedings. 1999 International Symposium on*, pages 109–112. IEEE, 1999.

[WVV+01] Holger Wache, Thomas Voegele, Ubbo Visser, Heiner Stuckenschmidt, Gerhard Schuster, Holger Neumann e Sebastian Hübner. Ontology-based integration of information-a survey of existing approaches. In *IJCAI-01 workshop: ontologies and information sharing*, volume 2001, pages 108–117. Citeseer, 2001.

[YAF+11] Cristiane A Yaguinuma, Gustavo F Afonso, Vinícius Ferraz, Sérgio Borges e Marilde TP Santos. A fuzzy ontology-based semantic data integration system. *Journal of Information & Knowledge Management*, 10(03):285–299, 2011.

[ZCWY14] Yu Zheng, Licia Capra, Ouri Wolfson e Hai Yang. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):38, 2014.

[ZCZ05] ZHU Zhuangsheng, Xiuwan Chen e Feizhou Zhang. Study on data fusiion method in geographic information system for transportation. In *Geoscience and Remote Sensing Symposium*, 2005.

[ZMW09] Laomo Zhang, Ying Ma e Guodong Wang. An extended hybrid ontology approach to data integration. In *Biomedical Engineering and Informatics, 2009. BMEI'09. 2nd International Conference on*, pages 1–4. IEEE, 2009.

[ZRC14] Ana Zaiat, Rosaldo JF Rossetti e Ricardo JS Coelho. Towards an integrated multimodal transportation dashboard. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 145–150. IEEE, 2014.

[ZZWP08] Tian Zhao, Chuanrong Zhang, Mingzhen Wei e Zhong-Ren Peng. Ontology-based geospatial data query and integration. In *Geographic Information Science*, pages 370–392. Springer, 2008.

BIBLIOGRAFIA

Anexo A

Anexos

De seguida encontram-se os anexos que se pretendeu incluir no trabalho, de maneira a complementar a informação já existente neste documento.

A.1 Método de Elaboração da Revisão da Bibliografia

Nesta secção é descrito o método que foi seguido para a construção da revisão bibliográfica. É também explicado ao pormenor cada uma das sub-etapas seguidas baseado no método descrito por *Arksey and O'Malley*[AO05], que descrevem no artigo do ano 2005 uma metodologia que visa facilitar a elaboração de revisões de literatura, chamando estas de revisão de escopo.

Esta metodologia sugere 5 tópicos a seguir para a elaboração desta revisão, e são estes: a definição de questões de investigação, pesquisa de artigos relevantes que tendem de alguma forma responder às questões propostas, seleção dos artigos a partir dos resultados de pesquisa que irão ser incluídos nesta revisão, mapeamento de conceitos relevantes para esta revisão com os artigos selecionados, e por fim a análise de resultados com tirada de conclusões e uma possível resposta às questões de investigação.

A.1.1 Desenvolvimento das questões-base de investigação

Conforme a metodologia de *Arksey and O'Malley*[AO05] indica, todos os aspetos da área a investigar devem tentar ser abrangidas pelas questões criadas. Para isso foram criadas 5 perguntas que tanto tentam focar na pesquisa da área de interoperabilização de múltiplas perspetivas de análise de dados como a junção de fontes de dados e a sua automatização. Por último é analisada a literatura que menciona a arquitetura orientada a serviços que também usa de alguma forma ontologia.

A seguir são apresentadas as questões:

1. Como interoperabilizar múltiplas perspetivas de análise de transporte/tráfego a partir de *Open Data*?

2. Qual o método de junção de fontes de dados a utilizar para uma eficiente obtenção e processamento de dados rodoviários (*data integration*; *sensor integration*)?

3. Quais os métodos de junção de fontes de dados, para a questão acima, tendo por base um modelo ontológico?

4. Como automatizar o processo de adição de uma nova fonte de informação ao já existente modelo ontológico?

5. De que forma será possível aplicar uma arquitetura orientada a serviços ao modelo ontológico que se pretende aplicar, de modo a permitir outras entidades acederem aos dados?

A.1.2 Estratégia de Pesquisa

Tal como foi dito em [AO05], existem variadas fontes onde poderá ser feita a pesquisa de artigos relevantes, tais como bases de dados eletrónicas, listas de referêncas, pesquisa manual em jornais e organizações relevantes como p.ex. conferências.

De modo a estreitar o espaço de pesquisa foram feitas variadas decisões quanto às fontes de informação e à maneira de a pesquisa ser feita. Para começar, foi decidido que o limite inferior das datas do artigos não poderia ser abaixo de 2002. Quanto às fontes de informação, a pesquisa foi somente feita em bases de dados eletrónicas, tanto por questões de tempo como por questões práticas. De maneira a não se perder tempo com tradução de artigos e linguagens que não o Português e Inglês, somente foram pesquisados artigos nestas duas línguas, tendo o foco principal ficado para a língua Inglesa pois hipoteticamente é aquela que trará a maioria dos artigos relacionados. É de referir que embora estas opções foram tomadas por questões práticas, poderão ter havido artigos relevantes para o trabalho que não foram incluídos por causa disso[AO05].

Quanto às bases de dados eletrónicas utilizadas, foram elas a Scopus[1], a Engineering Village[2], Web of Science[3], ACM[4], Google Scholar[5], Proquest[6] e Aleph[7].

Para a pesquisa foram identificados termos pertinentes que de alguma maneira poderiam descrever o escopo do problema que se pretende resolver. Estes são: *Ontology, transportation as a service, GIS-T, sensor integration, data integration, spatio-temporal data, open data, interoperability of analysis tools, multi-perspective, interoperability, multi-domain, collaborative*. A partir destes termos foram criadas *queries* para cada uma das questões apresentadas na Secção 1.3, a serem usadas nas bases de dados referidas anteriormente. São elas:

Pergunta 1

[1] http://www.scopus.com/
[2] http://www.engineeringvillage.com/
[3] http://webofknowledge.com
[4] https://dl.acm.org/
[5] https://scholar.google.pt
[6] http://search.proquest.com/
[7] http://aleph.fe.up.pt/

- interoperability AND transportation AND analysis
- interoperability AND transportation AND “open data”

Pergunta 2

- transportation AND data AND integration
- data AND source AND integration AND transportation
- (data OR source) AND integration AND method
- (data OR source) AND integration AND method AND transportation

Pergunta 3

- ontology AND data AND integration AND transportation
- ontology AND data AND source AND integration
- ontology AND interoperable AND transportation

Pergunta 4

- automatic AND source AND ontology
- automatic AND ontology AND integration

Pergunta 5

- service AND oriented AND arquitecture AND ontology
- SOA AND ontology

Em termos da 6ª questão apresentada, essa é a pergunta que se pretende responder com o trabalho desenvolvido.

A.1.3 Seleção de Artigos Relevantes

A partir dos resultados devolvidos pelas variadas bases de dados utilizadas, obtiveram-se muitas vezes resultados na ordem dos milhares, noutras vezes menos frequentes na ordem das dezenas, e muito pouco frequentes na ordem das unidades. Para que fosse possível extrair a maior parte dos artigos que realmente são importantes para este trabalho, usou-se a ordenação dos resultados por relevância que cada uma das bases de dados providencia aos seus usuários. Desta forma, foram visualizados no máximo 50 artigos por pesquisa podendo este valor ser menor, dependendo da pesquisa efetuada. O sorteamento destes artigos era feito inicialmente pelo seu título e de seguida caso este artigo despertasse interesse era analisado o seu *Abstract*. Posteriormente caso o artigo demonstrasse ser útil para este trabalho, era guardado para posterior análise.

A.1.4 Mapeamento de Conceitos

Sendo este passo da metodologia um processo iterativo, à medida que os artigos foram sendo analisados, os mais importantes foram sendo adicionados ao capítulo 3 que retrata os trabalhos relacionados.

A.1.5 Sumário e Síntese de Resultados

O propósito desta fase final é a de fornecer uma estrutura para a literatura descoberta. Devido ao abrangente foco da pesquisa e ao relativamente grande volume de literatura encontrada nas pesquisas realizadas nas bases de dados, sintetizou-se esta etapa final numa demonstração de cada um dos artigos relevantes num texto expositivo.